

### Supervision and control XML-based from Windows to Windows CE

## Tutorial: Getting Started with Movicon 11

Version 11.4 - Ed. July 2014 cod. DOCS 11 TUT-E Build 1151

## **Table Of Contents**

1. GETTING STARTED WITH MOVICON	3
1.1. INTRODUCTION TO THE MOVICON TUTORIAL	
2. PROJECTS	5
<ul> <li>2.1. How to create and structure a project</li> <li>2.2. Creating a New Project</li></ul>	6 9 .10
3. TAGS	13
<ul> <li>3.1. How to create Tags</li></ul>	15 16 18
4. SCREENS	23
<ul> <li>4.1. How to create a Screen</li> <li>4.2. Screen at the project startup</li></ul>	25 26 27 29
5. DYNAMIC ANIMATION	33
<ul> <li>5.1. How to create Dynamic Animations</li></ul>	33 36 39 39 39 42
6. ALARMS MANAGEMENT	-
<ul> <li>6.1. How to Manage Alarms</li> <li>6.2. Inserting Alarm Objects</li> <li>6.3. Displaying Alarms</li> <li>6.4. Displaying Alarm History</li> <li>6.5. Create a Simulation</li></ul>	45 50 52

## **1. Getting Started with Movicon**

#### **1.1. Introduction to the Movicon Tutorial**

Welcome to the Movicon tutorial. This tutorial is aimed at giving you a quick guided demonstration of the main Movicon Scada/HMI platform techniques used. At the end of this tutorial you will have learnt the most essential techniques for using Movicon base functionalities.

Before going ahead with this tutorial you should first install the software by using the setup procedures.

All the information in this document is based on the assumption that:

- 1. Windows is the operating system being used
- 2. The user knows how to use the Windows' techniques
- 3. The user has sufficient knowledge on automation systems, on variable and PLC concepts

For further information on each argument, please consult the Online Guide or the User's Manual

# **2.1. How to create and structure a project**

Starting up Movicon with the option command line, the program will start in Programming mode (Developer). The last project being used is usually opened. The workspace will display empty upon the first execution. The workspace uses the modern disappearing window techniques and therefore just simply point the mouse on the Tab you require to make it appear in the workspace. To keep the window displayed in the workspace use the relevant commands as indicated below:



Note: to display your working windows just point them with the mouse and use the dock command to keep them visible.



Movicon Workspace with hidden windows



Dock command



Movicon workspace with window kept displayed



Tip: you can close the property window and double click on it to make it re-appear.

#### **2.2. Creating a New Project**

To create a new project, use the 'New' command from the File menu (Ctrl+N).



A Wizard will appear to guide you in creating the new project:

					? >
ıt					
-	6			**	
Win32 platform	Web Browsers (j2se)	WinCE platform	Mobile Phones (j2me)	Empty Project	
-					
Template project					
				F	
					Open Cancel
	platform	Win32 platform Web Browsers (j2se)	Win32 Web Browsers platform (j2se) Template	Win32 Web Browsers platform (j2se) Template	Win32     Web Browsers     WinCE     Mobile Phones     Empty Project       Win2trom     (j2se)     Platform     (j2me)     Empty Project

First of all you need to select the type of platform on which the project must be run. In this way the functions which are not supported by the selected platform will not be available in programming mode (the selection can be changed later). Confirming this operation will display the configuration window:

Project Name		X
	Please, enter the name and the path where your project will be saved	
	Name Test	
	Folder C:\Temp\Test\Test	•
	<ul> <li>Crypt Core Project file</li> <li>Crypt all Project Resource files</li> <li>Compress all the files</li> <li>Encode using Unicode UTF-16</li> </ul>	
	< Back Next > Cancel	Help

In the window (as shown above), you will need to enter the desired name of the project in editing phase.

The other settings are not to be used for the time being but can be checked out in the manual if wished.

Click on the Next button to open the 'Users' settings.

	Password Protec			leveloper name her	e>
	Developer Pass	-	******	*****	-
	re-type Developer Pass	word	*************		-
	Enable Password Mng Create Default User Grou Create Users from Windo Enable Runtime Users ch Enable Windows User Lo Enable CFR21-Part 11 Se	wsName anges ogin	2	Enter Server Name	h

The security settings can be defined in this window. We will skip this part for the time being and go on with Next button to access the Driver settings.

Applicom Bacnet	^	Property	Value	
Beckhoff B&R	≡			
🖕 CAN Open				
CEI-ABI Duemmegi				
ELAP Elkron				
EL.MO	~			
				<u>×</u>
<				>

The drivers you wish to include in the project can be selected from this window. We will also skip this part and leave it for later. We will not set anything at this stage. Now click on Next button to reach the Screens Settings.

Add Screen Caption
Add Screen Navigation Bar
Default Screen Width 1280

Here you can indicate whether or not to create screens in the project. You can also indicate whether to create each one with a Title, and a contents navigation bar with scroll page buttons on the bottom border.

The default setting can be left alone or adapted to your requirements which can always be changed later.

Going still ahead with the wizard other windows will be shown for the configuration of the possible historical, alarms etc..

On the last window, Alarm Settings, when confirming these operations with Finish button, the wizard will proceed creating the project according to the settings carried out.



The Wizard will create the project's structure by pre-setting all the basic configurations in automatic.

#### 2.3. Workspace

By default, the Movicon workspace appear as shown below:





The Movicon Workspace, windows, toolbars, properties can be full customized.

#### **2.4. Project Properties**

Each Movicon project has properties, which are used to set all the project's configuration functions. To display the project's properties, click on the project name, at the beginning of its tree structure, or select the name and activate the Properties Window with the right mouse key.

ilter		·		
rojects		•		
Resources $\nabla$	Prope	erties		<b> </b>
🖃 🙀 Test	<b>O</b> T	est Project		
🕀 🎩 Alarm List (Nr. Alarms '3', N				
🕀 📕 Basic Scripts	and the owner where the party of the local division of the local d	X   🔡 🐴 🛢 루   Q	I	
🖃 🧮 Data Loggers And Recipes		General		
tog10min		Platforms		
tog10sec		Project Paths		
📻 Log1min		Execution		
🔚 Log30sec	9	Startup Screen		
📻 Log5sec		Startup Script		
-	9	Shutdown Script		
	9	Startup Commands	1 4 4	
	9	Shutdown Commands	<b>F</b>	
		Pre-Load Screens		
		🛯 Start F 📶 een		
<b>\</b>	K	🖊 Shew Stelus Bar		
		Show Output Window		
		E Advanced		
	🕀 F	leap Memory Manager		
	9	Set Values from the last Run		
	l. I	Memory Used with the curre	0	
	F	Recalculate memory Used		
		📃 Enable Heap Values		
	H	Heap Rectangles	0	
	H	Heap Alarm Winds	0	
	H	Heap HisLog Wnds	0	
	H	Heap DataLogger-Recipes	0	
	H	Heap Trace Winds	0	
		-	-	 
	Adv	vanced		

The Project's properties permit you to setup the general characteristics of the project itself, among which are:

- 1. Eventual encrypted file protection
- 2. Selecting project's destination Operating System
- 3. Working Folder paths
- Setting Startup behaviour (runtime execution- includes the 'Enable Renaming Manger' for automatically renaming variables linked to objects)
- 5. Operating system access security
- 6. Heap Memory settings for CE
- 7. Historical Log settings
- 8. Spooler print settings



For further details on all the properties please refer to the Programmer's Manual.

#### 2.5. Project Structure

The Movicon projects are built from a set of files XML format. Each project resource is saved in a XML file in the relating project's folder and in the subfolder of the relating resource.

Unless specified otherwise, the projects are saved in the "Documents \Movicon Projects" default folder.

The files, being 'open' thanks to the XML, can be encrypted and compressed in the project by means of using the project's properties. The structure of the files respects the structure the resources provided in the Movicon project window.

Let's go over the structure of the project files in detail, using the Windows Resource Explorer.



### 3.1. How to create Tags

To introduce a new variable (Tag) into the project you need to:

1. Select the Real-Time Database resource from the project window

Project Explorer			ųΧ	
Filter			-	
Projects			*	
Resources	$\nabla$			
🖃 🙀 Test				
🕀 🔔 Alarm List (Nr. Alarms	s'			
🕀 📕 Basic Scripts				
🕀 🚍 Data Loggers And Re	:ci			
🕀 💽 Event Object List				
🕀 🏶 List Child Projects				
⊞ Menus				
Hetwork Services				
🛛 🕀 🛒 OPC Client DA (COM)	1			
🕀 🗒 Parameter Files				
🖃 🔆 Real Time DB				
List Comm.Driver:	s			
📲 List Structure Pro	it			
🖹 <u>List Variables (Ta</u>				
🕀 🖃 Scaling Object List	1	New Variable	(Tag)	and the second second
🗉 🕀 🚯 Scheduler Object List	-*	New Valiable	Group	
Screen Navigation Ed	lit 📬			ariable (Tag)
E Screens		14644 5	dd a new Va	riable (Tag)
🖃 🥂 Shortcute	12	Add N		

2. Select the "Add a new Variable" command from the Command Pane found at the bottom of the project window. You can also use the analog command by using the right mouse key.

Resources	Туре	Area Type
🖃 🙀 Test*		
🗉 🔔 Alarm List (Nr. Alarms '		
🗉 💼 Basic Scripts		
🗉 🗮 Data Loggers And Reci		
🗉 🛐 Event Object List		
🗉 🎭 List Child Projects		
🗉 🔳 Menus		
🗉 🖛 Network Services		
🗉 🛒 OPC Client DA (COM)		
🗉 📕 Parameter Files		
🖃 🗱 Real Time DB		
🔊 List Comm.Drivers		
🖺 List Structure Prot		
🖃 🛃 List Variables (Tag		
⊕ 🐷 <u>VAR00001</u>	Word (1	Not Shared
🕀 🔄 Scaling Object List 🛛 🔓		

A new variable will be created in the project with default name and properties. The Properties Window, if hidden, is displayed by double-clicking on the new variable (if can be further displayed by using the relevant command from the 'View' menu).

You now need to assign the properties deemed necessary, especially the General properties, through the 'Properties Window'.

In our case we shall keep the default settings, with the PLC address to be assigned later. However we shall briefly go over the main properties for you:



Let's go over which are the fundamental properties of each Tag:

Name: permits you to assign the name desired for the variable.

Type: permits you to specify the data type (bit, byte, word, etc.)

**Area**: permits you to indicate whether an explicit memory area is to be used for the supervisor. When leaving the area as 'Not Shared', the supervisor will decide if the tag must be considered for the licence. The tag will be counted for the licence only if it's exchanged with the field through the driver, OPC, etc..

**Dynamic Address**: permits you to set the physical address to connect to the Tag to. The Tags Explorer can be used to specify the connection by means of an I/O Driver, OPC or Networking.

 All the other properties allow you to go and specify the Tag's behaviour, in the project, in detail. We, therefore, advise you to refer to the Programmer's Manual for further details.

We will leave the Tag with its default settings for the time being.

#### **3.2. How to Communicate with Drivers**

New communication drivers (I/O Drivers) can be inserted into the project at any time. In order to do this you need to:

1. Select the Real-Time Database Resource from the project window



2. Select the 'Add a new Comm.Driver' command from the Command Pane found at the bottom of the project window. You can also use the analog command by using the right mouse key.

3. A window will appear through which you must choose the driver you need from the list of drivers available.

4. Each driver is subdivided into product categories. By clicking on one product will get you the drivers and the relative communication protocols available.

Ethernet S7-200-300-400 TCP	Prop	erty	Value	
Ethernet S7-300/400 TCP	🗆 G	ieneral		
S5-AS511 Cpu Port	N	ame	PC Adapter	
S5-DK3964	Fi	ileName	MpiPcAdapter.	.dll
S7-MPI Hilscher NetLink Ethernet	V	ersion		
S7-MPI PC Adapter	L	ast Error		
Simatic Net SAPI S7				
SNMP	∃			
Vipa	~			
				,
upported protocol:S7-MPI protocol upported devices: Siemens PLCs S7 300	) and 400 and		a and compatible day	

Check the communication driver relating to the product and the protocol desired.

• We will check the Siemens S7 MPI "PC Adapter" protocol for our example.

When confirming the operation the driver will be inserted into the project and added to the list of drivers in the project window.

We can now proceed with necessary configurations through the properties window:

Pro	perties		ąх
ø	PC Adapter Driver		-
~	X 🔡 🐴 🖹 🗭 I	0   T	
Θ	General		
	Name	PC Adapter	
	FileName	MpiPcAdapter.dll	
	Version		
	Last Error		
	Settings		
	Check for Updates		
Ð	Features		
-	Symbol Libraries 🖉 👘 Dyn	amic Help	_

First of all you must proceed with the driver settings configurations from the General properties group.

Go to the 'Settings' item where you will find an activation button for accessing the communication settings window.

#### **3.3. Configuring the Driver**

In this example we have chosen to use the Siemens S7-MPI PC Adapter driver as an example. The techniques used are the same for all the other drivers accept a few protocol specifications. The first thing to do is sort out the configurations of the driver's General Characteristics.

	operty MPI Network Settings	Name
_	Network Bitrate	187.5 KBit/sec
	MPI Address	1
	Highest Address	31
	Only Master	1
Ξ	General	
	Wait Time	0
	Timeout	2000
	Minimun Threshold	5
	Aggregation limit	0
	C 1011	

1. Usually the default settings are left as they are accept for certain specifications required by the device being used. As an example lets suppose we have a standard PLC with a standard MPI connection for which we will keep the General default settings.

2. After the general settings, select the 'Stations' window needed for the communication station settings which we will create for the driver.

Add 🕂	Name		
Edit Edit			
🛥 Remove			
Test Cable/Comm.			
Edit the list of Stations.	-		
This feature allows to enter and define the Station list			

3. Use the "Add" button to add the necessary communication station to the driver in order for it to communicate.

4. When entering the new Station, its relating settings window will display through which we will configure the communication details of our station for which we will only concentrate on the fundamental properties.

Ξ	Device Station Settings	· · · · · · · · · · · · · · · · · · ·
	Station ID	2
Ξ	General	
	Station Name	Default Station
	Error Threshold	1
	State/Command Variable	
	Keep Opened	True
Ξ	Serial Port Settings	
	Port	1
	Baudrate	38400
	D + 01	

**Station Name**: Assign a name to the station. In our case we will put PLC1 (but any other name is acceptable).

**Port**: Assign the serial port number being used. In our case we will use the COM1 serial port, for which we will leave the value left at 1.

**Baudrate**, Byte Size, Parity, Stop Bit: Assign the parameters of the communication port. In our case we will keep the Default settings. Station ID: this is the last property on the list whose setting is based on the ID address set in the PLC.



All the other station properties permit you to further configure the communication modalities. For instance, the TAPI functions can be used for communicating via modem or the Bridging functions used for communicating via the modem on the PC, to use the same communication port for the PLC's remote maintenance (eg. Teleservice). To get further information on these features please consult the Programmer's Manual.

However, we will limit ourselves in using just the base functions relating to device communication for the time being.

When confirming the settings, the communication station will be inserted in the communication driver.

Other stations for communicating with other devices on different COM ports can also be inserted with the same MPI protocol.

MpiPcAdapter	
General Stations	Tasks About
-∰ Add 1≊ <mark>f</mark> Edit	Name PLC1
🗕 Remove	
Edit the list of Stations. This feature allows to enter and define the Station list	
	OK Cancel Apply Help

When arriving at this point the driver should have been inserted and the device already connected and ready for communicating. To verify whether all is in order and working correctly we shall run a test by using the "Test Cable/Comm." button. In this way Movicon will be able to verify whether communication with PLC device has been set up correctly and the cables are correct. Any errors found should then be resolved to ensure that communication works correctly.

## **3.4. Assigning Physical Addresses to Tags**

After having inserted at least one station, we will look at how physical addresses are assigned to Tags.

1. Select the Tag previously inserted into the project (or create a new one)

- Image: State of the state
- 2. Double-click on it to open the Properties Window.



3. Select the 'Dynamic' property from the 'General' group to open the Tag Browser window.

Network 🙀 OPC			
List Comm.Drivers in the Project	Comm.Driver Proper Remove	Property	Value
🗉 💋 PC Adapter			

4. Select the Tab relating to the communication driver from the Browser window.

5. Double-click on the PC Adapter previously inserted to open a window to assign the physical address.

Device Address     DB1.DBW0       General     Station       Station     PLC1       Conditional Variable     Input/Output       Type     Input/Output       # Elements     0	
Station     PLC1       Conditional Variable     Input/Output       Type     Input/Output       # Elements     0	
Conditional Variable       Type       Input/Output       # Elements	
Type Input/Output # Elements 0	
# Elements 0	
Write outputs at startup False	

6. Select the driver station with which you wish communicate with (in our example we have only entered the station named PLC1), then specify the device's physical address in the "Device Address" to which the variable is to be connected.

7. In our example, we shall connect the Word type variable called VAR00001 to the PLC's DB1 data block's word DW0.



Note: You can also enter the syntax of the physical address In the Tag's `Dynamic' property directly:

[DRV]PC Adapter.Sta=PLC1|Addr=DB1.DBW0

With the Tag property set, Movicon will establish communication with the device for reading-writing data from the PLC on the corresponding variable during project runtime.

#### **3.5. Importing Tags directly from PLC**

The Movicon drivers offer an extremely useful feature when the database of ready-made PLC variables is being used:

The '**Import-Update device database'** command, from the Command Pane, is made available when selecting the driver from the Movicon project window. This same command can be obtained and used by clicking the right mouse key.

When activating this command you will be request to select the file (keeping the CTRL key or SHIFT key pressed down) corresponding to the PLC database. As we are using Siemens S7 we need to select the .SDF or AWL file by means of the file selection window:

Open					? 🗙
Look in:	Cimatic S7		•	🗢 🗈 💣 🃰	-
CO Recent	🖻 S7-Exp-examp	ple.sdf			
Desktop					
My Documents					
My Computer					
<b>S</b>	File name:			•	Open
My Network	Files of type:	System Data Files (*.sdf)		•	Cancel
Places		Open as read-only			

When selecting the file with the PLC database, the Movicon Import Device variables window will open to allow you to select all or part of the variables contained in the PLC database.

ort Device Variables - C:\Samples\Simatic S7\S7-Exp-example.sdf		
Please select the Device Variables to Import		
009_M1_EnableForBefDCC - M165.7 - BOOL	~	Select All
2_Tobacco - M315.5 - BOOL		Select All
AATR3 - MD706 - REAL		
ABypassSafetyBwd070M1 - A0.6 - BOOL		Select None
ABypassSafetyBwd080M2 - A1.0 - BOOL		
ABypassSafetyBwd090M2 - A1.2 - BOOL		
ABypassSafetyFwd070M1 - A0.7 - BOOL		Browse File
ABypassSafetyFwd080M2 - A1.1 - BOOL		
ABypassSafetyFwd090M2 - A1.3 - BOOL		-
Active_Check_Cicle - M303.0 - BOOL		?
Acustic_Alarm - A2.5 - BOOL	_	
AMTC4 - MD822 - REAL		
AMTC5 - MD826 - REAL		
AMTR4 - MD710 - REAL		
APSR3 - MD702 - REAL		
ATTR4 - MD714 - REAL		
auxFNMotor020Run - M159.5 - BOOL		
auxFNMotor130Run - M159.4 - BOOL		
auxOneShotStartBatchOB35 - M335.0 - BOOL		Import
Avg(1) - MD874 - DWORD		Impore
Avg(2) - MD878 - DWORD		
Avg(6) - MD882 - DWORD	~	Cancel
Ava(7) - MD886 - DWORD	×	
Station: In or		
Station: PLC1	-	

When confirming the operation the 'Import' button, Movicon will go ahead with:

1. Creating the Tags in the Movicon project keeping the same name and type taken from the PLC database

2. Assigning the relative physical address to each Tag

By using this useful function you can get the Movicon project's Variables DB created and completed with the device's physical addresses assigned automatically in just a few seconds. Each Tag's 'Dynamic' property will be shown associated with the following syntax (which can be changed as pleased):

[DRV]PC Adapter.Sta=Default Station|Addr=M265.0|Typ=0

-

#### 4.1. How to create a Screen

To create a graphic interface you need to used the project's Screen resource.

1. Select the **Resources Folder** from the Project window's tree structure.

Resources	V	
🖃 🔯 Test*		
🗉 🔔 Alarm List (N	Nr. Alarms '3', Nr. Runtim	
🕀 🔚 Basic Scripts	5	
🕀 🚍 Data Logger	rs And Recipes	
🕀 💽 Event Objec		
🐁 List Child Pro	ojects	
🕀 🔳 Menus		
🗉 <table-cell-columns> Network Ser</table-cell-columns>		
🕀 🛒 OPC Client [		
🕀 🧱 Parameter F		
🗄 👯 Real Time D		
🕀 🔄 Scaling Obje		
🗄 🕔 Scheduler O		
Screen Navi	igation Editor	
	0.000	
Shortcuts	Open	
E ASoft Logic	📸 Add a new Menu	
Costis Alia	🕸 Add a new Shortcut	
122	📰 Add a new Script	en e
	🖄 Add a new Screen 👘	Script Explorer
	🖄 Add a new Folder	
ommands	📕 🛛 Add a new Parameter	
📸 Add a new Menu	V	Insert a new Screen in the Project
	M Cut	

2. Select the '**Add new screen in the project**' command from the Command Pane at the bottom of the project window. You can also use the analog command by using the right mouse key.

Q			Movicon - [Design] - Sc	een1				_ = X
Elle Edit Yew Layout Symbols Icols	Window Help							
	-1816	0. 🕹 🐺 🕄	8 B B 8 8 5 1	E QIAIO				
Project Explorer	a ×	Screen1 ×			-	Properties		a × m
Filter	•		1 - 3 - 1 - 4 - 1 - 5 - 1 - 6	1 • 7 • 1 • 8 • 1 • 9 • 1 •	.0.1.1	Screen1* Screen		
					-	V MIRIA BI	POT	2
Projects		-				⊖ General		8
	Objects	-				Name	Screen1	
😑 🙀 Test"		_				ID IN ARRE	G	
🛞 🔔 Alarm List (Nr. Alarms '3', Nr. Runtim	1100010000	à				Width	1280	
🛞 🚊 Basic Scripts						Height	1024	
Data Loggers And Recipes		[·]				Advanced	1024	
Event Object List	10000	m.				Style		
List Child Projects		-						
		4				Background     Execution		
		-				* Execution		_
Image: State of the state of		i.			18			
🛞 🧱 Parameter Files		1 ·						
Real Time DB								
Scaling Object List		<u>°</u>			100			
Scheduler Object List		-						
Screen Navigation Editor		~						
B Screens		-						
	0(0)							
* Shortouts		·						
Soft Logic		1.						
Users And User Groups		0.						
		-						
		ò						
		1 -						
		L:				22.1 20.2 (ADD 30.00) 20.2 (A		
		2						
	Þ	-			×	Concession of the local division of the loca		
and a second	12100000		Lange and the second		>	Symbol Libraries	Dynamic Help AProper	ties
Commands		Script Explorer		a x a	Refactoring Ex	plorer		Φ×
🛱 Add New Screen Local Variable				and the second se				
Add New Variable Script Event	1000	V 🖾 Insert -	N S = E A O		多1四美			
Create a WebClientX html page based on this 5	Screen	Object: (General)	<ul> <li>Proc: (decla</li> </ul>	ations)				
📑 Add a new Menu		and the second sec	,					
4 Add a new Shortcut		1		>				
				1				
Insert a new Basic Script in the Project	100010001	H I H H Scree	en1 /		HAPH	Screen1		
E Insert a new Screen in the Project		IL Logic Explorer	Script Explorer		XML Code	e Explorer Refactoring	Explorer	and the second se
Cutput								
For Help, press F1			CAP NUM :	and later in the later is a second	-	10. 11.83	(1,16b)	and a second second
For nep, press ni			[[Cap]NOM]:			1.936	(1,100)	i. 194

3. The new screen will be created in the project and displayed with its default settings in the workspace.

4. You can change the screens default properties through the **Properties Window**. This window is accessed by double-clicking on the screen itself or by using the same command from the View menu.

5. We will only deal with the screen's background color properties. Please refer to the Programmer's Manual for details on the other properties.

6. Select the **'Back Color**' property from the **'Background'** property group and assign white as the screen's background color.

This property takes effect when being confirmed with the  $\checkmark$  key.



7. Repeat this operation to introduce another screen into the project. By doing this we can setup an example to be used in the 'change page' lesson up ahead.

#### 4.2. Screen at the project startup

To get the screen to open automatically at the start of a project Runtime you need to specify the screen in the **project's Execution properties.** 

Project Explorer	<b>д &gt;</b>
Filter	-
Projects	*
Resources	$\nabla$
🗆 🔯 Test* 📐	
🗉 🔔 Alan SList (Nr. Alarms '3', Nr.	Runtim
🕀 🔳 Basic Scripts	
🗉 🕀 🔁 Data Loggers And Recipes	
🕀 🗈 🚺 Event Object List	
List Child Projects	
⊞ Menus	
🕀 🛃 Network Services	
🕀 🐨 🖉 OPC Client DA (COM)	
🕀 🧱 Parameter Files	
🕀 🐺 Real Time DB	

1. Double-click on the project name at the beginning of the project's tree structure to display the its properties, or select the name and activate the Properties Window with the right mouse key.

Pro	perties	<b>д &gt;</b>
Ø	Test* Project	•
~	X 🗄 🖓 🗗 🛑 🛛	0 T
۰	General	
Ð	Platforms	
Ð	Project Paths	
	Execution	
	Startup Screen	Screen1
	Startup Script	
	Shutdown Script	
	Startup Commands	<b>F</b>
	Shutdown Commands	<del>7</del>
	Pre-Load Screens	
	📃 Start Full Screen	
	📝 Show Status Bar	
	📃 Show Output Window	
	Advanced	
Đ	Heap Memory Manag	
Ð	<b>Historical Log Settings</b>	
Ð	Spooler Print Manager	
	<b>cartup Screen</b> ows you to enter the Screen I	to be loaded at startup [ID12144]
The second	Symbol Libraries	amic Help 🔧 Properties

2. Select the **Execution Property**, then the **'Startup Synoptic**' property. By using the activation button, activate the window for selecting the screen desired. Then press the **'Refresh'** button to add it to the list.

Resource Browser		
4 Screen		Þ
Filter		Refresh
Screen1		7
•		
	OK	Cancel

3. We will select `Screen 1' to use in our example (or you can select another one if you prefer). Then confirm with OK.

The specified screen will be the one to open and display automatically at the project startup.

#### 4.3. Graphic Editing

We will now re-open 'Screen1' to examine the basic graphic editing concepts.

1. Double-click on the 'Screen1' resource, found in the Screens folder in the project window, to open the screen.

2. Use the drawings tools by taking them from the  $\ensuremath{\text{Toolbox}}$  positioned on the workspace's right border.

3. Select the '**Basic Shapes**' from the Toolbox and then select the drawing to be used graphically on the screen.

	Mov	icon - [Design] - Screent	and the second secon		_ = ×
Elle Edit Yew Layout Symbols Looks Window	i Help				
- E 🖬 🗑 🖶 🗛 🔛 🐘 🛍 🖛 • 🖘 • 🍂	I III, I O, I & III (III) (III) (III)	1 🕼 🎶 📾 🌾 🛝 🔍 📥 1 🚱			
Project Explorer	a × Screent ×		-	Properties	Toolbox + ×
Filter	* +1+1+1+2+1+3+1	+4+1+5+1+6+1+7+1+8+1	9 - 1 - 0 - 1 - 1 - 1	Test* Project	E Basic Shapes
Projects	<ul> <li>4</li> <li>4</li></ul>		<[]	Connectal     Connectal     Connectal     Connectal     Connect Patho     Connect Patho     Statup Screen     Statup Screen     Statup Connecta     Statup Connec	Image: Point of the point
is 愛Screen1* 電 登 Schrouts 聞 Schrouts 電 後Ueers And User Groups	· · · · · · · · · · · · · · · · · · ·		×	Spooler Print Manager     Startup Screen     Allows you to enter the Screen to be to     Symbol Libraries     Oynamic H	
Commands	<ul> <li>Script Explorer</li> </ul>		a × Refactoring Expl	orer	
Continues           Continues           Toporto           New Project           Conclusion and Open Device Project           Conclusion the Cross Reference           New Child Project           Upload Project to Device/PTP			HIIIMA	fest /	
For Help, press F1		CAP NUM SCRL :	) <u>11</u>	(1,8Gb(1,1Gb)	1 1 181.

4. After having selected the chosen drawing, double-click on the insertion point on the screen and drag the drawing until you reach the size desired.

5. Repeat these operations to create the drawing you want on screen.

6. These graphic elements, once on the screen, can be given general, style and animation properties by using the **Properties Window** as described below.



#### **4.4. Object Libraries**

In addition to the Basic shapes provided in the Toolbox you can also access other graphic object categories. These categories contain vectorial drawings with style and animation properties similar to

Toolbox		Toobon	
Teolibes - Basic Stapts - Seat Chess - Battons Lights - Seat Chess - Order - Order - Radis Ration - Radi	Toullous	X Toolbas "     Basic Stapes     Basic Stapes     Basic Stapes     Basic Stapes     Basic Stapes     States Light-Switches     States Chart-Data Analysis     Mono Exter     Work Deter     Work     Work Deter     Work     Work     Work     Work	Toolbox     T
Televie Button     Bite Button     Green Button     Green Button     Interprety     Selector 8     Selector 8     Selector 8     Space Bute     Space But     Space But	Call Top Heter  Call Top Heter  Call Tot Network  Call Regist Meter  Call Regist Meter  For Vencto Banif auch  For Vencto Banif auch  For Vencto Gang graph  For Vencto Gang Gangen  Call Danie Gaugen  For Banif Gangen  For Banif	Advanced Shapes	RR         Extension Conjung           Table Workp         Extension Conjung           Stable Workpe         Extension Conjung           Rest Active Workpe         Extension Conjung           Data Logger Rest Predow         Extension Conjung           Track Store Workpe         Extension Conjung
Square Yellow Square Green Square Block Sladers-Gauges-Meters-Displays Trend-Charts-Data Analysis Advanced Shapes	Trend-Charts-Data Analysis     Advanced Shapes		G Scheduler Window 100 <sup>4</sup> IP Canera Verver

those of the basic shapes but already predisposed with execution functions for which they have been designed for.

To use the **ToolBox's Objects**, simply select the object desired then double-click the mouse to insert the object on the point of the screen where you want it to be and drag it until you reach the size desired.



After having inserted the your chose objects, you can then proceed with assigning their properties by using the **Properties Window**. Each object will have, apart from the general properties, also style and animation properties, which are common to all objects, and the execution properties specified for each single object.



Please refer to the Programmer's Manual for further details.

#### 4.5. Symbols Library

Movicon provides a vast variety of graphic symbols in libraries which have been pre-built purposely to meet all the graphical representation requirements in automation.

These symbol libraries can either be accessed through the **'Symbol Libraries**' window, displayed on the border on the right hand side of the workspace, or by using the analog command from the 'View' menu.



Each symbol from each category can be inserted on screen by simply using the Drag&Drop techniques and re-sizing it as desired by dragging its borders just like any other graphic object.



The symbols can be configured in their properties just like any other drawing object, by using the Properties Window.

#### Graphic editing example:

Insert a 'Rectangle' object into the screen from the 'Basic Shapes' ToolBox and a 'Tank' symbol from the Symbol Library.



#### 4.6. Creating a Composed Symbol

All the drawing elements (Drawings, Symbols, Objects) can be grouped together in Symbols and then added to the Symbol Library.

Now let's proceed with inserting a few drawing elements which we will then associate to a graphic symbol.

By following the procedure described above, insert a Rectangle and two Ellipses from the Basic Shapes ToolBox to form the shape shown below:



Select all three elements with the mouse by clicking in the area and dragging the selection. The figure below shows how the drawing should look like with the **reference object** highlighted for any eventual align commands.



With the right mouse key, in the workspace, select the **Symbol – Group** command to group all three drawings together to make one symbol.



The symbol can now be added to the Movicon Templates library by using the right mouse key on **Symbol -> Add** to Library.

Any animations or codes associated to the symbol will also be kept in the library.

The composed symbols can be inspected in the project's tree structure. The objects and the composed symbols are displayed in structures under the screen they belong to.

Therefore each components of each symbol can be selected singularly and configured in its properties.





When using this technique we suggest you assign a name to each symbol or drawing so that they can be identified straight away.

### **5. Dynamic Animation**

#### **5.1. How to create Dynamic Animations**

We will now look at the editing techniques used, which entail the association of Tags, for creating dynamic animations.

#### **5.2. How To create Dynamic Colors**

In our example we have chosen to assign the animation properties to color the drawing's background in function with the VAR00001 tag previously introduced.

- 1. Activate the screen where the graphic symbols were inserted as described above.
- 2. Select the **rectangle** shape representing a tube.
- 3. Double-click or use the other techniques to display the Properties Window.
- 4. Select the Animation group from the Properties Window and then the Back Color item.



Check the **`Enable**' box to enable the pre-selected animation function, then select the tag among those inserted in the project's RealTime DB. Then select the **`Edit back color list**' to set the **activation thresholds** of the tag and the relating

colors to be displayed.

A window for setting how the Thresholds should appear.

Ellipse3 - Bac	kground C	olor		
Color Threshold				
Color		/ari		Add
	1,000 0,000			Edit
				Remove
				Default
			-	Derdak
				Сору
				Paste
		ОК	Cancel	?

The window contains a series of standard default thresholds. Use the relative commands on the side to delete, add of edit them.

Use the relative settings window, shown below, to add or edit the animation characteristics:

Property	Value
Value for Threshold Color	1
Variable for Threshold Color	i i i i i i i i i i i i i i i i i i i
Visualization Mode	normal
Blink Time	500
Color Lime (00ff00)	
Blink Color	White (fffff)

By using this window you can set the threshold values and the animation colors, as well as other properties which are explained in the Programmer's Manual. Confirm the settings with OK.



When you **Run the project**, changing the real-time value of Tag VAR00001, you will see the rectangle shape change color.

Attention: the "Variable for Threshold Color" in this threshold settings window consents to making the activation threshold dynamic. This variable, however, MUST NOT be used in the 'Background color' property, otherwise the color animation will not work correctly by showing only the same color without changing.

**D** 

TIP: Variables can be associated to objects by directly dragging them from the RealTimeDB resource and dropping them on the objects on screen.



When you select a variable from the variable list and drag and drop it on an object in the screen, a window will display allowing you to select which animations to associate to that variable.

Assign to Dynamic for Polygon2	- VAR00002	$\mathbf{X}$
<ul> <li>Visible</li> <li>Background Color</li> <li>Text - Edge Color</li> <li>Move Horizontal (X)</li> <li>Move Vertical (Y)</li> <li>Composed Movement (XY)</li> <li>Scaling</li> <li>Gradual Filling Color</li> <li>Gradual Filling</li> <li>Rotation</li> </ul>	OK Cancel	

However, the threshold color settings remain at the user's discretion.

#### **5.3. Other examples of Dynamic Animations**

We will now insert some animations requiring Word type tags, which can be inserted into the project as described above in chapter 4.

Let's assume that two tags, VAR0001 and VAR00002, both in Word are available in our example project.

We will demonstrate another example of dynamic animation for on screen symbol movement: **Composed Movement**.

1. Open the screen and select the Symbol, created previously with the **Rectangle** and **Ellipse** drawings grouped together, then activate the **Properties Window**.

2. Select the **'Animations'** group and then **Composed Movement**. This animation sets the graphic symbol to move on the screen along a trajectory line graphically drawn out with the mouse in proportion to the tag values associated.

- 3. Check the **'Enable'** box.
- 4. Associate the VAR00002 tag previously inserted

Propert	ties		ά×		
💽 Syr	🗐 Symbol4 Symbol				
~ >	< I 🎛 🐴 📑 루 I 🤅	9   T	=		
🖯 Dj	ynamics		-		
Đ	Visible				
Ð	Transparency				
	Composed Movem				
	📝 Enable Composed				
	Variable	🔋 VAR00002			
	Start Value	0			
	End Value	100			
	Edit Path				
	Reset Path				
Đ	Move Horizontal (X)				
(H)	Move Vertical (Y)				
		et the position of the object 571]			
LE S	ymbol Libraries 🖉 Dyna	mic Help 🔧 Properties			

5. Confirm with 🔨.

6. Close the property window and activate the mouse's right key commands from the selected symbol. Select the 'Edit Composed Movement' item.

7. Drag the symbol's shape to the end point, i.e. to the right hand side of the tank.

8. To insert the intermediate points of the path, double-click with the mouse on the line and drag it to the intermediate point desired and continue like this **until the trajectory line is complete**.

9. Press the **ESC** key when finished. You should get this result as shown below:


10. Select the  $\ensuremath{\textbf{Scaling}}$  box from the Animations properties group to activate the relating settings window.

11. Associate the VAR00002 tag previously inserted.

12. Enter the 50-100 values as scale Percentage, so that the symbol remains visible at 50% of its scale as minimum value.

13. Select the direction of the scaled re-sizing (leave the default selection).

Properties	<b>ф</b> ×
🗐 Symbol3 Symbol	•
🗸 🗶 🔡 🗛 📑 📮	10 T
Start Y Point (Siz	:e) 🔺
🕀 End X Point (Size	•]
🕀 End Y Point (Size	=]
😑 Scaling	
📝 Enable Dynamic	S
Scaling Variable	👸 VAR00003 📃
Start Value for Scalir	ng O
End Value for Scalin	g 100
Min. Percent Scale	50
Min. Percent Scale	100
Scale Direction	+ê+ center
Botation	
🕀 Text - Display Va	lue
😑 Background Colo	I
📃 Enable Backgrou	un
Variable Back Color	
📃 Variable Is Alarm	G 💌
	um percent value for the scale sizing ou to force a minimum scale size even
Symbol Libraries	Dynamic Help 🔧 Properties

#### Confirm with



When you **Run the project**, changing the real-time value of Tag VAR0002, you will see the movement through the path of the symbol. The size (scaling) of the object will change consequently.



TIP:Variables can be associated to objects by directly dragging them from the RealTimeDB resource and dropping them on the objects on screen.



When you select a variable from the variable list and drag and drop it on an object in the screen, a window will display

allowing you to select which animations to associate to that variable.



However, the threshold color settings remain at the user's discretion.

### **5.4. How to execute commands from Objects**

Execution commands can be assigned to objects in the screen's user interface, in function with their characteristics.

We will follow up the chapter reported above regarding graphic animation to complete the argument by explaining the techniques used for assigning execution commands to object.

In our case we shall use a '**Button'** object and a '**Gauge'** object, which are needed to produce the animation which we configured previously.

1. Activate the screen where the graphic symbols, described above, are inserted.

2. Take a 'Gauge' from the ToolBox's "Sliders-Gauges-Meters" category and insert it in the screen.

3. Take a 'Green Button' from the ToolBox's 'Buttons-Lights-Switches' category and insert it in the screen.



### **5.5. Assigning Tag's value from Objects**

#### **Example Using Button Objects**

Let's proceed with configuring the **button** which we will use for acting on the variable used for managing the rectangle's (tube) color animation.

1. Double-click or use the other technique to display the inserted Green Button's Properties.

2. Select the **Execution** group from the properties window and then the **'ON-OFF' Mechanic Style**. Select the VAR00001 tag previously inserted. By using this characteristic the button will toggle the Tag, by setting it with the '0' and '1' values. The tag can also be interacted on by using the command selection as we will show you up ahead.



3. Confirm with  $\checkmark$ .

#### **Example Using Gauge Objects**

Now we shall configure the gauge which we use to interact on the tag to manage the created symbol's animated movement.

1. Double click or use the other technique to display the inserted gauge's Properties Window.

2. Select the Variable group from the properties window and then select the **Gauge-Slider Variable** item. Select the VAR0002 tag previously inserted. In this way the gauge will interact directly on the VAR00002 tag. The gauge object is totally configurable, by using the numerous properties provided. It is only necessary to do the configuration as indicated for our example. The other properties can be referred to in the Programmer's manual.

Properties	Ţ	×
💭 Gauge1 Gauge		•
X B 4 9 0	T	-
😑 Variables		-
Gauge-Slider Variable	👸 VAR00001	
Scale Settings		
Default Struct	B	
Advanced		
🗉 Style		
Gauge Type	circular	
Knob Style	💎 Style 3	
Slider Color	3DSHADOW SysColor (	
Knob Color	WINDOW SysColor (ffffff)	
Bar Back Color	3DSHADOW SysColor (	
Bar Fill Color	MENU SysColor (ffffff)	
Bar Brush Style	Solid	
Scale Color	BTNTEXT SysColor (00	
🕀 Gauge Circular Setti		
Gap	5	
Show Slider		
Show Title		
Show Bar		
Show Scale		
Scale Right-Bottom		-
Gauge-Slider Variable Enter the variable (Tag) associated value [ID12373]	d to this Gauge-Slider as dynamic	>
📖 Symbol Libraries 🎧 Dynami	ic Help 🔧 Properties	

#### 3. Confirm with



TIP: you can also drag the Tag from the project RealTimeDB resource directly to the object on the screen, to simply assign the variable.



If you select a variable from the variable list and drag it on top of an object on screen, it will automatically be inserted in the command object's Tag property.

### **5.6. Assigning executing commands to Objects**

Different types of commands can be assigned to any command object (Buttons, Menu, Accelerators) and command lists can also be created. The commands can be activated by selecting the **"Command Type"** on the **Execution Properties** as **"Execute Command"**, then defining the command type by selection the **"Command on Release"** or Command on Pressed.

The button's execution properties are:

VAR00001 Execute Commands		-
VAR00001 Execute Commands		-
VAR00001 Execute Commands		•
Execute Commands		_
Execute Commands		
£		
7		
4		
·		
	-	
🗆 Simple		
🔍 green light button		
🗆 Null		
3DFACE SysColor (e0df		
None		
(ffffff)		
WINDOW SysColor (ffffff)		
		-
e button have to execute		
	<ul> <li>green light button</li> <li>Null</li> <li>3DFACE SysColor (e0df)</li> <li>None</li> <li>(ffffff)</li> <li>WINDOW SysColor (ffffff)</li> </ul>	green light button  Null  3DFACE SysColor (e0df  None  (ffffff) WINDOW SysColor (ffffff)  window sysColor (ffffff)

When activating the **'Commands'** selection from the Execution properties you can edit the command list to be associated to the object by using the **'Add New Command**' button in the **Command List** window.

Command List	X
	New Command
	Edit
	Remove
	Tip : Commands are executed in the list order.
	Drag and drop items in the list to change the execution order
	OK
	Cancel
	?

The '**Add New Command**' button opens the settings window of the operating commands to be assigned to the object. Each configured command will be added to the Command List which the object will execute.

Action Move To Variable Value Strobe Time (ms)	<b>VAR00001</b> Set <b>1</b>
Move To Variable Value Strobe Time (ms)	1
Value Strobe Time (ms)	1
Strobe Time (ms)	•
Max.Value	0
	100.0
Min.Value	0.0
Max.Chars	0
Value	

There are commands in Tags (Set, Reset, Toggle, Strobe, Increase, Decrease, Virtual Keyboard...) or on Screen windows (with the various opening modalities).



Please refer the manual for further details on all the command operations which can be assigned to objects.

### 5.7. Start Runtime

At this point, with the objects configured we can run the project to verify its Runtime behaviour.

1. Press the Markov button or use the Start Project command from the File menu (or ALT+F12).

2. Movicon will ask you to save the project. Save the project using the classic Windows techniques.

3. After having saved the project on file, it will be executed in run mode where you can operate the objects to see if they work.

4. To return to Programming mode, use the ALT+F12 keys or the button from the bar. Note: (you can customize system menus by inserting all the commands desired as well as for the Movicon or Windows shutdown from the project in Runtime mode).

## 6.1. How to Manage Alarms

In this brief lesson we will quickly see how to activate, display and record alarms in Movicon projects. We shall continue with our example from where we left off with a few Tags and a pair of screens already predisposed in our project.



Note: The alarms are objects from the project. Each alarm has their own General properties where they are assigned names and associated to tags in cases when not used as templates. Alarms used as templates will be dealt with further on.

Each alarm is built with at least one threshold, whose value and condition determine the activation of the alarm with an associated text.

## **6.2. Inserting Alarm Objects**

1. Select the 'Alarm List' Resource from the project window which in turn will show the relative commands in the command Pane at the bottom.

2. Use the **"Add a new Alarm"** command from the Command Pane or with the right mouse key. A new alarm object will be created in the project and can be renamed as pleased.



Activate the alarm object's properties window to assign the desired name, i.e. ALL001, then the Tag from the project by selecting it from those inserted in the project's Real-Time DB. In our example, the object's properties are those indicated in the figure below:

Properties	ά X
🛐 Alarm Alarm	•
X 🔡 A E 🗭 0	T
😑 General	
Name	ALL001
Device Name	
Alarm Variable	👸 VAR00001
Advanced	
Enable Alarm Variable	el B
🔲 Variable Duration	
Enable Dispatching Mes	en al construction and a constru
Hysteresis Alarm Value	0
Exclusive Thresholds	
🔽 Enable Alarm	
Enable only if Quality	
Alarm Variable Enter the variable (Tag) for this ala	rm [ID12309]
Symbol Libraries	ic Help 🔧 Properties

Important: if the 'Quality Good Only' is left checked the alarm will be activated only when the RealTime DB assigns the tag with a certain value. For instance, in cases where a Tag connected to a driver or in network, becomes disconnected the value turns to 'uncertain' and therefore the alarm will not appear. If in doubt, uncheck this option for a test run.

3. After having entered the Alarm object, you can enter at least one **activation threshold**. Therefore, select the alarm from the project Window and use the '**Add a new Alarm Threshold**' from the Commands Pane or use the right mouse key.

Project Explorer	џ×
Filter	-
Projects	
Resources	
🖃 🔯 Test*	
🖃 🔔 Alarm List (Nr. Alarms '4', Nr. Runtim	
🖃 🛐 ALLOO1	
Threshold	
🕀 🕎 Analog Alarm	
🕀 🕎 Digital Alarm	
🕀 🛐 Digital Message	
🕀 📕 Basic Scripts	
🗉 🗮 Data Loggers And Recipes	
🕀 🚺 Event Object List	
🗉 🙅 List Child Projects	
🕀 🔲 Menus	
Hetwork Services	
Ø OPC Client DA (COM)	
Parameter Files	
Commands	*
🖄 Add a new Alarm Threshold	
Add a new Alarm	
-	

4. By doing this the alarm will show one intervention threshold which we will configure through its properties.

5. We will configure the alarm's activation on the value desired in the threshold properties Window.

6. Go to the 'General' properties group to assign the **'Title'** being the text which will be associated to the alarm. The title can be typed directly in the property box or, as a good rule, can reside in the project's **'String Table'** and there may be subject to language change.



Note: when using the string table, you need to select the project name from the project resource window and use the 'Edit String Table' command from the Command Pane (or using the command made available for use with a right mouse click). Then insert the columns (each column is a text language) and then proceed with inserting the texts which will then be made available all over the project

7. We then have to assign the threshold value in the **'Value'** box in the **'Execution'** property group. You can also use a 'dynamic' threshold value, where the alarm activation value derives from the contents of another tag. Leave the default activation >= (more than or equal to).

8. The Style and Notification Event properties are of no interest to us for the time being and therefore we will leave those for default.

• The alarm is historically logged in the file for default and can be traced or reset as well as other characteristics to be referred to in the Programmer's Manual.

Properties	ά×
Threshold Threshold	•
× XI 🔡 🐴 🖹 부 I Q	T T
General	
Threshold Name	Threshold
Alarm Area	Theshold
Alarm Text	
Alarm Help	
Duration Message Format	
Read Access Area Level	FFFF
Write Access Area Level	FFFF
Execution	
Activation Value	0
Activation Condition	major-equal
Severity	1
Delay (sec.)	0
Advanced	
Alarm Threshold Variable	Te de la companya de
Commands on CTRL+ d	4 4 4
Commands when Alarm	<del>7</del>
Commands when Alarm	Ŧ
Commands when Alarm	<b>F</b>
Commands when Alarm	<del>7</del>
⊖ Style	
Support ACK	
Support RESET	
Allow RESET with Condi	
🗹 Blink	
🗹 Print	
Record on Historical Log	
V Beep	
TextColor	Automatic
Advanced	<u>•</u>
Advanced	
Symbol Libraries 🕐 Dynami	c Help 🔧 Properties

The alarm and its activation threshold are now configured. You need to consider that each alarm may have different activation thresholds and if the associated variable is not bit type, but Word type for instance, the alarm is consider to be analog type.

This procedure permits one alarm to be created, with different threshold if need be, for each variable. However, there is another way that allows you to set alarms as **"Templates".** In order to do this you need to set the alarm as described above, **but without specifying the name of the associated variable.** 

This will make the alarm generic and associated to more than one variable at the same time. If you set more than one variable the List Variables, they can be selected at the same time by pressing the SHIFT or CTRL key.



Once you have selected the variables, you can associate an alarm using the "Associate an alarm" command from the command pane at the bottom or using the right mouse key to get to it.

List Structure Protot		
🖃 🛃 List Variables (Tags)	(Tags	7, La
	_	
🕀 👿 VAR00006	-01	Associate a DataLogger/Recipe
🕀 💀 VAR00005	0	Associate an Alarm
🕀 🐷 VAR00004	101	Associate an Event
🕀 🐷 VAR00003	100	Associate this Tag to an Alarm
🕀 🐷 VAR00002		Synchron Associate this Tag to an Alarm
🕀 🗔 VAR00001	X	Cut
🕀 🔄 Scaling Object List		Сору
🕀 🕔 Scheduler Object List		
👬 Screen Navigation Editor		Paste
🖃 🛄 Screens	2	Properties
🗉 🛄 Screen2*		
🕀 🛄 Screen 1*		Compile Cross Reference
I Thortcuts		Source Control

This command allows user to select the alarm from the previously defined alarm list.



Note: It is always best to differentiate the type of alarm you intend to use, therefore when that alarm is used as template with multiple variable associations, you should make sure that the alarm has not been set with a variable in its "Alarm Variable" property.

We will now define a new alarm called ALL002 without associated it with a variable (Tag).

Resources	$\nabla$	Device	Variable	Enable Variable	Exclusive	-
🗆 🙀 Test*						
😑 🔔 Alarm List (Nr. Alarms '2', Nr. I	Runtim					
🗉 🛐 ALLOO1		-	VAR00001	-	No	
🗆 <u>🛱 ALL002</u>		-	-	-	No	
Threshold	5					
🗉 🚛 Basic Scripts	Ŷ					
Data Loggers And Peripes						

We shall then add one variable named VAR0003 to the list of variables as described previously. Now we can select both the VAR00002 and VAR00003 variables from the list using the CTRL key technique and call the **"Associate an Alarm"** command with the right mouse key.

	⊕ 💹 <u>VAR00003</u>	Word (1 No
Ľ		

We shall go ahead and choose the ALL002 alarm. We will then see listed the single alarms associated to the two variables. In this case the alarms will behave exactly in the same way as the ALL001 alarm does, simply knowing that the VAR00002 and VAR00003 variables have identical alarm thresholds, even though logged individually for each variable.

Filter	
THUCH	-
Projects	
Resources V	•
⊞ Menus	
E 😰 OPC Client DA (COM)	
🕀 📕 Parameter Files	
🗆 🙀 Real Time DB	
🗊 List Comm.Drivers	
E List Structure Prototypes	
🖃 🛃 List Variables (Tags) (Tags 7, La	
🕀 🚾 VAR00007	
🕀 🚾 VAR00006	
WAR00005	
VAR00004	
VAR00003	
🕰 ALLOO2	
🕀 🛅 Variable Used in	
🖃 🚾 VAR00002	
ALL002	
🕀 🗄 Variable Used in	
E Scaling Object List	
🗉 🚯 Scheduler Object List	
Screen Navigation Editor	
E Screen1*	•
c 1	
Commands	_
Associate this Tag to a DataLogger/Recipe	
Associate this Tag to an Alarm	
f Associate an Event to this variable	
Synchronize SoftLogic Variables	
🙀 Add a new Variable (Tag)	
Add a new Variable Group	
<u>↓</u>	

The alarm and its activation threshold (each alarm can have a number of activation thresholds) have now been configured.

We can now move on to how to view active alarms and those historically recorded on file.

# **6.3. Displaying Alarms**

The active alarms, setup in the project's Alarm List resource, can be displayed in purpose-made object viewers which can be inserted on the screen.

We need the use of a screen. In our example project we have setup two screens, 'Screen1' and 'Screen2' where Screen1 has already been used for the graphic examples. Therefore we are left with Screen2 for this example.

1. Double-click on 'Screen2', in the Screens folder from the project Window, to open it in edit mode.

ect Explorer a X	Screen?      Screen? x     Properties	0
	1 1 1 1 1 2 1 1 2 1 1 3 1 1 4 1 1 5 1 1 6 1 1 7 1 1 8 1 1 9 1 1 0 1 1 1 1 2 1 1 2 1 1 3 1 1 4 1 1 5 1 1 6 1 1 Street 2* 5 1 1 0 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1	0
		BPIO
ojects ·	S General	
Resources	Name	Screen2
* Menus	ID.	0
8 +**Network Services	Viet	1290
* CPC Clent DA (COM)	Hecht	1024
8 E Parameter Files	# Advanced	
B @Real Time CB	Style	
Ext Comm.Drivers	Keep alma	
I List Structure Prototypes	Y Fit in Wind.	
Br List Variables (Tags) (Tags 7, La	Advanced	
8 🖬 VAR00007	Background	
# # VAR00006	Back Color	White (MM)
* 🖬 VAR00005	Background L	
	Image X por	0
ALL002	Image Y pos	0
* Tel Variable Used in	Close Screen	5000
E Variable Used In     E Variable Used In	* Advanced	
ALL002	® Execution	
* Te Variable Used in	Screen Layer	8
a g variable used in a.	Public Source	
* Siscaling Object List	Style Source	
* Scheduler Object List	Advanced	
Screen Navigation Editor		
R Screens		
* Screen?*		
* Screent*		
# to shotes		
da		
Milbar Antibar Groups		
emmands .		
Associate this Tag to a DataLogger/Recipe		
Associate this Tag to an Alarm		
Associate an Event to this variable		
Synchronize SoftLogic Variables	Image X pos	
Add a new Yariable (Tag)	Enter the X-collect	x the Bk Image
🗳 Add a new Variable Group	[012148]	
Add a new Comm. 1/O Driver		Dyma A Prop.

2. Activate the Toolbox and take out an 'Alarm Window' object from the 'Advanced Shapes' category.

3. Click on a point on the screen's top left, then drag the selection to insert the 'Alarms Window' object in the size desired.



4. Double-click to activate the 'Alarm Window' object property. The **Style** property permits you to fully configure the Alarm Viewer object. The **Background** property permits you to assign the background colour desired for the alarm's window. The **Font** property permits you to choose its characteristics as desired. The numerous properties, described in the Programmer's Manual, permit you to manage the viewing of alarms according to any applied necessities. In our case, for simplicity, we will leave all the default settings as they.

пх

* 🗶   🔡 🐴 📑 📮   🕢	1.1
Style	-
Border	🔳 sunken
V Auto Column Layout	
Show Control Window	
Ack Sel Button	
Ack All Button	
Reset Sel Button	
Reset All Button	
Toggle Sound Button	
Help Button	
Get History Button	
Advanced	
Clickable	
View Expanded List	
Button Size	small
Align Buttons	bottom
Ack Sel Button Text	
Ack All Button Text	
Reset Sel Button Text	
Reset All Button Text	
Toggle Sound Button Text	
Help Button Text	
Get History Button Text	
Description Column Name	
Alarm On Column Name Alarm Ack Column Name	
Alarm Off Column Name Alarm Beset Column Name	
Alarm Duration Column N	
Alarm Total Time ON Col	
Alarm Severity Column N Alarm Status Column Name	
Alarm Condition Column	
Alarm Image Column Name Time Format	
i ime Format	
dvanced	



You can modify, add or take away the columns describing the alarms in the Alarms Viewer by using the appropriate tools, which are displayed with the "Shift + Double-click' in the same window.

Alarm Description	ſ	Time ON		Duration
	Fiel	ld Choice 🛛 🛽 🖸		
	Т	ime ACK	T I	
	Т	ime OFF		
		ime RESET		
		tatus		
		nage otal Time ON		
		otar nine on		
Ack All (Ctrl+A)	Reset Sel			Toggle Sound (S)

# **6.4. Displaying Alarm History**

All the alarms are historically logged for default. The recording modalities and the Historical Log archives management can be customized through the Historical Log properties which is accessed by selecting the project name from the tree structure and then using the properties Window.

To display the historical data of the alarms, you need to proceed as described above for the Alarms Viewers.

Apart from the Alarm Viewers you will also find the Historical Log Viewer in the ToolBox. Carry out the same procedures for inserting the Historical Log as described above for the Alarms Viewer.

Keep in mind that the Historical Log window displays system messages for default only and not alarm messages. In order to display alarm messages in our project we will have to set the Log Window's 'Filter Event Type' property to the "All" or "Alarm Messages" value.

		on - [Design] - Screen2		The second s			-
Elle Edit Yew Layout Sygpols Iools Window Help							
	L   O,   🌣 🗣 🖓   🗞 i 📰 i 🕼 🌾	11日 21日 21日 11日	0				
* 25 葉 林   風 風   鹿 日 風       *     🦰							
oject Explorer 3 ×	Screen1 Screen2 ×				-	Properties	Ģ
ilter •	+1+1+1+2+1+3+1+4+1+5	. 1 . 6 . 1 . 7 . 1 . 8 . 1 .	9 • 1 • 0 • 1 • 1 • 1 • 2 •	1 • 3 • 1 • 4 • 1 • 5		Screen2* Screen	
rojects ·	:	Alarm Desc	rintion		<u>^</u>	~ XI # A	
Resources C		Aldini Dest	anpoon			☺ General	
⊕ ≡ Menus	1					Name	Screen2
C. Although Constant						ID	0
OPC Client DA.(COM)	c.					Width	1280
Parameter Files	-					Height	1024
Real Time D8	~.					Advanced	
List Comm.Drivers	-					B Style	
	4					Keep alwa	
Er List Variables (Tags) (Tags 7, La	-					Fit in Wind	
The second seco	i.					Background	
* 🗖 VAR00006						Background Back Color	White (mm)
R VAR00005						Back Color Background I	White (mm)
	· .					Background I Image X pos	0
E VAR00003	-					Image X pos Image Y pos	0
ALL002	e-					Close Screen	
It Variable Used in	-					Advanced	
VAR00002	Ack Sel (A) Ack All (Otri	+A) Reset Sel (R)	Reset All (Ctrl+R)	Toggle Sound (S)		Execution	
🙀 ALL002	- ACC SEL(A) ACC ALL (COS	FA) Reset Set (R)	Resec Al (COHR)	roggie sound (s)		Screen Layer	
Tel Variable Used in	o						
W WAR00001	Ever	nt Text	Event Tir	me User	Des		
Scaling Object List						Advanced	- Torrestore
	<u>e</u>						
Screen Navigation Editor							
G Screens	-					100000000000000000000000000000000000000	
Screen2*			16				
Screen1*	in land						
* 2 Shortcuts	-						
Soft Logic	<u>.</u>						
ommands *	+						
Associate this Tag to a DataLogger/Recipe	4						
Associate this Tag to an Alarm							
	<u></u>						
	11						
Synchronize SoftLogic Variables						Image X pos	
📱 Add a new Yariable (Tag)	Refresh (P5)	Fib	er (F2)	Print	(P)	Enter the X offset for IID121481	r the Bk Image
📫 Add a new Yariable Group	I.					(1012140)	
🕰 Add a new Comm. I/O Driver					×		
·· · · · · · · · · · · · · · · · · · ·	- < ]				>	Symb/ C D	yna 🔧 Prop
Output Script Explorer	actoring Explorer						
For Help, press F1	and the second		[]CAP[NUM]SCRL] 6,35	I DE LOCALE		1,856(1,266)	

All we have to now is verify what we have done. For this we need the following function in the project, considering all that has been realized up to this point:

- 1. A command for alarm simulation on the alarms page.
- 2. The change page commands.

### **6.5. Create a Simulation**

We shall find room in the Alarms screen window to insert a command object which will interact on the VAR00001 tag associated to the alarms.

1. Open 'Screen2' from the project window. Arrange the viewer objects so that the is enough space left for inserting the other objects (i.e. on the bottom border).

2. Insert a **'selector'** object from the Toolbox and position it on the bottom border. Activate its properties and assigned the VAR00001 tag in the Execution properties.

	U T	
Execution		<b></b>
Variable ON-OFF.	👸 VAR00003	
Command Type	ON-OFF	
Commands On Release	÷	
Commands On Pressed	Ŧ	
Advanced		
🗉 Style		
🔽 Clickable		
Border	Simple	
Style	🐵 selector B	
Background Attributes		

The selector object will interact on the VAR00001 tag, which we have already assigned to the Alarm object.

The same can be done with the remaining VAR00002 and VAR00003. Therefore we shall insert another two selectors and assign one with the VAR00002 variable and the other with the VAR00003 variable.

3. Insert another new **'button'** object, again from the Toolbox, as before. Position it on the bottom border at the side of the selector. Activate its properties and assign the opening of 'Screen1' in the Execution properties. This button will then permit us to execute a page change to return back to the first page.

Property	Value
Screen	Screen1
Action	Open normal (screen change)
Monitor	0
Parameter file	
× Position	100
Y Position	100
Width	0
Height	0
Z Caption	
Ø Border	
Resize border	
📃 System Menu	
Maximized Box	
Minimized Box	

4. Now we can insert the same object into 'Screen1', the startup screen, so that we can dispose the open Alarms page command. The button object's 'Cut & Paste' can also be used to dispose it on another screen after which its properties can be modified.

#### **Executing Runtime**

We now have the necessary items arranged in our example project to test run it:

- Screen1: graphic simulation screen, with command and graphic animation objects. The appropriate button is used for accessing Screen2.
- Screen2: Alarms simulation screen, with the alarms activation and viewer objects.

At this point, we are all set for executing a test run of the project to verify its behaviour during Runtime.

1. press the 💟 button or use the Start Project command from the File menu (or ALT+F12).

2. Movicon will ask you to execute a project save. Execute the save according to the usual Windows' techniques.

3. After having saved the project file, it will be put into run mode letting you try out the objects to see if they work.

4. To return to Programming mode use the ALT+F12 keys or the **Markov** button from the bar.



Note: (you can customize a system menu by inserting all the commands desired, including the ones for shutting down Movicon or Windows from the project in Runtime mode). Please refer to the Programming manual for further details.

Alarm Description	Status	Time ON	Duration	Severity	Condition
Air Vacuum	ON	12/03/20		1	ON
VAR00002 - Threshold	ON	12/03/20		1	ON
VAR00003 - Threshold	ON	12/03/20		1	ON
Ack All (Col+A) Event Text	Reset All (O	Itri+R) Toggie Sound ( User	5	Desc	ot receives
AR00003 - Threshold	2008-03-12 15:20:48	036			eshold : VAR00
AR00002 - Threshold	2008-03-12 15:20:48				eshold : VAR00
AR00003 - Threshold	2008-03-12 15:20:42				eshold : VAR00
Air Vacuum	2008-03-12 15:20:45				hold : VAR00001
AR00002 - Threshold	2008-03-12 15:20:41				eshold : VAR00
Air Vacuum	2008-03-12 15:20:41				hold : VAR00001
AR00003 - Threshold	2008-03-12 15:17:15				eshold : VAR00
Threshold	2008-03-12 15:17:13				hold : VAR00001
AR00002 - Threshold	2008-03-12 15:17:13				eshold : VARODUI
ARUUUU2 - Threshold	2008-03-12 15:17:13			VARUUUU2 - Thi	eshold : VARUU
Refresh (F5)	Piter (	72		Print (P)	1
				Print (P)	
001 VAR00002	VAR00003				

The picture shows an example of our simulation.

In order to generate and delete alarms using the three selectors switches, you can get the alarm's history from Alarm Window as well. You can also analyse the alarm's history displayed in the Alarm Window from when it occurred using the "Get History (G)" command. If we select an alarm occurrence and click on the "Get History (G)" command, a '+' symbol will

If we select an alarm occurrence and click on the "Get Hisotry (G)" command, a '+' symbol will appear at the side of the alarm in the window. This symbol is used for expanding the occurred alarm's history.

Alarm Description	Status	
🛕 Air Vacuum	ON	12
A VAR00002 - Threshold	ON	12
VAR00003 - Threshold	ON	12
ALARM ON		20
🗉 😼 ALARM ON		20
ALARM ON		20

You can also check alarm occurrences in the Historical Log window and any other following operations carried out to them.

	Event Text	Event Time	User	Description	^
	VAR00003 - Threshold	2008-03-12 15:20:48		VAR00003 - Threshold : VAR0	
	VAR00002 - Threshold	2008-03-12 15:20:48		VAR00002 - Threshold : VAR0	
-	VAR00003 - Threshold	2008-03-12 15:20:42		VAR00003 - Threshold : VAR0	
	VAR00003 - Threshold	2008-03-12 15:20:45		VAR00003 - Threshold : VAR0	
	VAR00003 - Threshold	2008-03-12 15:20:44		VAR00003 - Threshold : VAR0	
	VAR00003 - Threshold	2008-03-12 15:20:43		VAR00003 - Threshold : VAR0	
	Air Vacuum	2008-03-12 15:20:45		ALLOO1 - Threshold : VAROOO01	
E 🔺	VAR00002 - Threshold	2008-03-12 15:20:41		VAR00002 - Threshold : VAR0	
E 🛕	Air Vacuum	2008-03-12 15:20:42		ALLOO1 - Threshold : VAR00001	
± 🛕	VAR00003 - Threshold	2008-03-12 15:17:15		VAR00003 - Threshold : VAR0	-
	Refresh (F5)	Filter (F.	1		





Movicon<sup>TM</sup> is a trademark of Progea, related to the HMI/SCADA platform entirely developed and produced by Progea. © 2012 All Rights reserved.

No part of this document or of the program may be reproduced or transmitted in any form without the express written permission of Progea.

Information in this document is subject to change without notice and is not binding in any way for the company producing it.



Via S.Anna, 88/E 41100 Modena - Italy Tel. +39 059 451060 Fax +39 059 451061 Email:info@progea.com Http://www.progea.com



Progea International Ltd via Penate 16 6850 Mendrisio - Switzerland tel +41 (91) 9676610 fax +41 (91) 9676611 international@progea.com



Via XX Settembre, 30 Tecnocity Alto Milanese 20025 Legnano (MI) Italy Tel. +39 0331 486653 Fax +39 0331 455179 Email: willems@progea.com



Progea USA LLC 2800 East Enterprise Avenue Appleton, WI 54914 Tel. +1 (888) 305 2999 Fax. +1 (920) 257 4213 info@progea.us

#### progea

Progea Deutschland GmbH Marie-Curie-Str. 12 D-78048 VS-Villingen Tel: +49 (0) 7721 / 99 25 992 Fax: +49 (0) 7721 / 99 25 993 info@progea.de