zenon manual

Multi-Touch

v.7.11

COPADATA
do it your way

Contents

1. Welcome to COPA-DATA help

GENERAL HELP

If you cannot find any information you require in this help chapter or can think of anything that you would like added, please send an email to documentation@copadata.com (mailto:documentation@copadata.com).

PROJECT SUPPORT

You can receive support for any real project you may have from our Support Team, who you can contact via email at support@copadata.com (mailto:support@copadata.com).

LICENSES AND MODULES

If you find that you need other modules or licenses, our staff will be happy to help you. Email sales@copadata.com (mailto:sales@copadata.com).

2. Multi-Touch

With zenon, touch screens can also be operated with Multi-Touch gestures.
You can obtain an example project (on page 15) from your zenon consultant.

REQUIREMENTS

The following is recommended for Multi-Touch:

- ▶ Use of a `DirectX 11`-compatible graphics card

- ▶ Activation of `DirectX hardware` for the zenon property `Graphical design`/`Graphics quality`

CONFIGURING MULTI-TOUCH

To use Multi-Touch:

1. Deactivate the `Windows CE project` project property

2. Configure Multi-Touch in:
   a) project properties
   b) screen properties
   c) properties of the dynamic elements

AMENDMENT IN PROJECT PROPERTIES

To amend the project property:

1. Go to the `Interaction` group

2. Select the `Recognition` function in the in `Multi-Touch` group
   - `Windows 7`: activates
     gestures that are supported by Windows 7, as well as Multi-Touch in VSTA Multi-Touch for zooming and scrolling in worldview
   - `Windows 8`: activates
     gestures that are supported by Windows 8, as well as Multi-Touch in VSTA Native Windows 8 Multi-Touch support

3. Select the desired properties in the other subgroups:
   - `Input visualization for pen`
   - `Input visulization for touch`

> ⚠ **Attention**
>
> *In the Control Panel of Windows 8, the visualization of the Finger and pen input can be set globally:*
>
> ▸ If the visualization is deactivated, then there is also no visualization in zenon, regardless of what has been configured.
>
> ▸ If `"Optimize visual feedback for the output to an external monitor"` is used, then the visual feedback is enhanced and always displayed in zenon, regardless of the settings in the project properties.
>
> *Default: `Visual feedback` is activated but no enhanced. The behavior can also be set with zenon.*

ADAPTATION OF SCREENS AND DYNAMIC ELEMENTS IN THE PROJECTS

In the project, you can find configuration options for Multi-Touch in the following properties:

▸ For screens, in the groups:
  - `Interaction`
  - `Programming interface`
  - `VSTA gesture recognition`

▸ For dynamic elements, in the groups:
  - `VSTA gesture recognition`
  - `Runtime/Tap and hold`

3. Gestures

With zenon Multi-Touch, you have all Windows 8 based gestures available. Above all:

Gesture	Description
`Scrolling (as with the mouse wheel)`	Touching a screen with a finger, followed by placing a second finger which is then moved.
`Double tap`	Two quick successive touches of an object
`Rotate`	At least two fingers touch the screen and carry out a circle movement clockwise or counter-clockwise.
`Tap and hold`	Touching an object that lasts for longer than two seconds.
`Flick`	Touching an object followed by a short, quick directional movement of the finger.
`Swipe`	At least one finger touches the screen and moves a short distance in the same direction.
`Tap`	A brief touch of an object with a finger.
`Move`	Touching an object followed by a liner movement of the finger.
`Zoom`	Moving two fingers away from one another.

4. Configuring the interaction

With Windows 8 gestures, interaction can be configured in many areas via Multi-Touch via properties with no additional programing necessary.

Requirement: In the project properties, `Windows 8` must be selected in the `Interaction` node for the `Recognition` property. zenon must be running on a Windows 8 computer in Runtime.

CONFIGURATION

Interactions can be configured for:

- ▶ Some screen types

- ▶ Dynamic screen elements

- ▶ Touchboxes

The actions that can be assigned to gestures depends on the screen and/or screen element that is to be configured.

To configure interactions for screens:

1. Set the focus to the desired screen

2. navigate to the `Interaction` node in properties

3. Select the desired properties in the subgroups

To configure interactions for screen elements:

1. Set the focus to the desired screen element

2. navigate to the `Runtime` node in properties

3. Select the desired properties in the subgroups

4.1 Reactions

Reactions to gestures can be individually configured for screens and screen elements:

1. The following are available in the `Interaction` group for supported screens:
   - Reactions to `Tap and hold`
   - Reactions to `Double tap`

2. The following are available in the `Runtime` group for dynamic screen elements:
   - Reactions to `Tap and hold`

3. In the `Runtime` group, the following are available for various elements, depending on the screen type:
   - `Double tap`
   - `Tap`

Reactions are used to define what happens when the respective gesture is detected on the screen or screen element in Runtime.

▶   `Reaction`: Selection of the desired reactions from the drop-down list. The reactions that are available depend on the screen type/element.

▶   `Function`: Selection of a function configured in zenon if `Execute own function` was selected.

Note: There is a significant difference between screens and screen elements with `Execute own function`: Interlockings and user rights can also be configured for screen elements. This is not possible for screens, because the screen does not have screen functions.

Screens

▶   can be used for runtime environments that are not touch-compatible

▶   are backwards-compatible: new gestures can also be supported and gestures can also receive new settings

If a screen is copied, the respective properties that have been set are accepted.

4.1.1 Manipulation

Screens and certain screen elements can be manipulated with touch gestures. You can define the desired reaction for a gesture in the Editor in `Manipulation` group for:

▶   `Move`: Extended Trend diagram window

▶   `Move horizontally`: Worldview, AML, CEL, touchboxes

▶   `Move vertically`: Worldview, AML, CEL, touchboxes

▶   `Zoom`: Worldview, AML, CEL, touchboxes, Extended Trend diagram window

4.1.2 Double tap

DOUBLE TAPPING ON SCREENS

The following screens are available as a reaction to `Double tap`:

▶   `No reaction`

▶ `Execute own function`

▶ `Display status window`

DOUBLE TAPPING ON ELEMENTS IN SCREENS

AML

The AML list in the `AML` screens aids as `Reaction` for `Double tap`:

▶ `No reaction`

▶ `Execute own function`

▶ `Acknowledge alarm`: for selected alarms

▶ `Execute alarm`: for selected alarms

▶ `Help for executing alarm`: for selected alarms

▶ `Stop/continue list`: independently of alarms.

Double tapping on a list entry always leads to this being selected and the corresponding function being executed. Double tapping in an area outside the list entries only leads to independent functions, but no alarm-specific functions.

EXTENDED TREND

The diagram window in an `Extended Trend` screen aids as `Reaction` for `Double tap`:

▶ `No reaction`

▶ `Execute own function`

▶ `Zoom to 100 %`

▶ `Rezoom`

LISTS IN OTHER SCREENS

Some lists in screens aid as `Reaction` for `Double tap`:

▶ `No reaction`

▶ `Execute own function`

4.1.3 Tapping with screen elements

The following are available for screen elements as `Reaction` on `Tap`:

- ▶ `No reaction`

- ▶ `Execute own function`

- ▶ `Selection`

4.1.4 Tap and hold

PRESSING AND HOLDING ON SCREENS

The following screens are available as a reaction to `Tap and hold`:

- ▶ `No reaction`

- ▶ `Execute own function`

- ▶ `Display status window`

PRESSING AND HOLDING ON SCREEN ELEMENTS

The following are available for screen elements as a reaction to `Tap and hold`:

- ▶ `No reaction`

- ▶ `Execute own function`

- ▶ `Display context menu`

5. Evaluating events

Events that are to be evaluated via the programming interface can be defined in detail for screens. The evaluation is carried out via VSTA at screen level. To configure the evaluation of events: the `Recognition` property in the project properties for `Interaction` must be set to `Windows 8`

1. go to group `Programming interface`

2. go to subgroup `Multi-Touch events`

3. Select the desired property from the drop-down list of the `Raw data event routing` property:

   - `All events`: all events are evaluated

   - `Deactivated`: the evaluation is deactivated

   - `Only selected events`: only the events activated via checkboxes are evaluated

The following events are available:

▶ `PointerDown`

▶ `PointerEnter`

▶ `PointerLeave`

▶ `PointerUp`

▶ `PointerUpdate`

▶ `PointerWheel`

▶ `PointerHWheel`

▶ `PointerDeviceChange`

▶ `PointerDeviceInRange`

▶ `PointerDeviceOutOfRange`

▶ `NCPointerDown`

▶ `NCPointerUp`

▶ `NCPointerUpdate`

▶ `PointerActivate`

▶ `PointerCaptureChanged`

You can find details about events in the Object model section or at the Microsoft Help for MSDN (http://msdn.microsoft.com/en-us/library/hh454903(v=vs.85).aspx (http://msdn.microsoft.com/en-us/library/hh454903(v=vs.85).aspx)).

6. VSTA gesture detection

The properties for VSTA gesture detection make it possible to activate and deactivate the flags of the interaction contexts separately. You can find the properties for the VSTA gesture detection in the properties for:

- ▶ Screens: Gestures that relate to the screen

- ▶ Dynamic elements: Gestures that relate to an individual dynamic element

The properties are only available if the `Recognition` property is set in the project properties for `Interaction` on `Windows 8`.

The selected configuration is available in Runtime and can be edited via VSTA.

You can find details about events in the Object model section or at the Microsoft Help for MSDN (http://msdn.microsoft.com/en-us/library/windows/desktop/hh448838(v=vs.85).aspx (http://msdn.microsoft.com/en-us/library/windows/desktop/hh448838(v=vs.85).aspx)).

SCREENS

You can find the properties for the VSTA gesture detection for screens in the `VSTA gesture recognition` properties group of the screen. As soon as the `Gesture recognition active` property was activated, the following gestures are available for selection:

- ▶ `Manipulation`

- ▶ `Exact`

- ▶ `Translation X`

- ▶ `Translation Y`

- ▶ `Rails X`

- ▶ `Rails Y`

- ▶ `Translation inertia`

- ▶ `Rotation`

- ▶ `Rotation inertia`

- ▶ `Scaling`

- ▶ `Scaling inertia`

- ▶ Cross slide

- ▶ Cross slide horizontal

- ▶ Cross slide select

- ▶ Cross slide speed bump

- ▶ Cross slide rearrange

- ▶ Cross slide exact

- ▶ Tap

- ▶ Tap double

- ▶ Secondary tap

- ▶ Drag

- ▶ Hold

- ▶ Hold (mouse)

DYNAMIC ELEMENTS

You can find the properties for the VSTA gesture detection for dynamic elements in the VSTA gesture recognition properties group of the element. As soon as the Gesture recognition active property was activated, the following gestures are available for selection:

- ▶ Manipulation

- ▶ Exact

- ▶ Translation X

- ▶ Translation Y

- ▶ Rails X

- ▶ Rails Y

- ▶ Translation inertia

- ▶ Rotation

- ▶ Rotation inertia

- ▶ Scaling

- ▶ Scaling inertia

▸ Cross slide Cross slide horizontal

▸ Cross slide select

▸ Cross slide speed bump

▸ Cross slide rearrange

▸ Cross slide exact

▸ Tap

▸ Tap double

▸ Secondary tap

▸ Drag

▸ Hold

▸ Hold (mouse)

7. Example project for Windows 7

This example project for Multi-Touch under Windows 7 is designed for a resolution of 1920x1080. It contains:

▸ Start page

▸ Navigation

▸ Alarm Line

▸ Screen of type Login

START PAGE

The start page displays an overview of the complete production line. Several equipment Icons are visible at the same time. You can scroll to other equipment using gestures. Tap on an Icon changes to the selected equipment. The following is also available in the screen:

▸ Alarm line at the top edge: Displays the last alarm of the complete production line. You can drag out the alarm line. This will display the whole Alarm Message List.

▸ Login button: Makes it possible to log in different users.

▶ Exit button: Closes the Runtime and can only be operated by users with administrator rights.

NAVIGATION

In the lower screen area, the navigation depicts the whole production line with the help of Icons in a horizontal scroll area. In addition an energy worldview is available. It is selected via the button located at the lower center. A machine is selected with a Tap on a visible Icon. In this project only the equipment `Filler` can be selected. If you press and hold the `Filler` Icon long enough, a Glow effect is displayed. The list can be scrolled using a Swipe gesture; a Tap on the scrolling list stops this again.

The scroll speed is determined via the acceleration of the Drag movement:

▶ Slow: follows the finger

▶ Faster: follows quickly behind

The navigation is centered on the `Filler` Icon when the start screen is called up.

ALARM LINE

At the top edge of the screen an alarm line is located. It displays the last alarm of the complete production line. It can be opened to display the Alarm Message List.
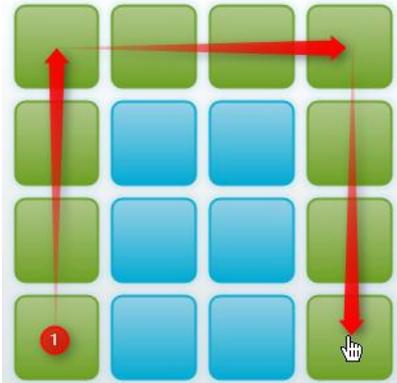Operation:

▶ Open:
  - A Tap on the line opens the AML up to the half of the screen.
  - Via gestures the AML can be customized to an individual size.

▶ Close:
  - A Tap outside of the frame closes the opened AML.
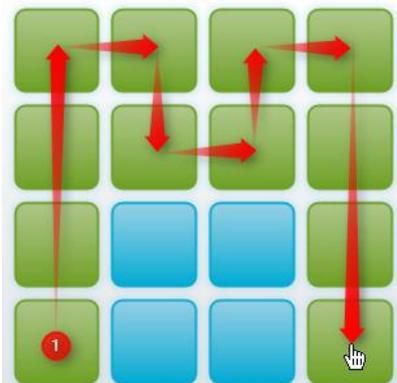 - You can also move up the AML manually.

LOGIN SCREEN

The Login screen offers a gesture-based login in the style of Windows 8. Before you enter a password, you must select a user by means of a Tap. After that you can start entering the password for the selected user via `Hovering`. For example:
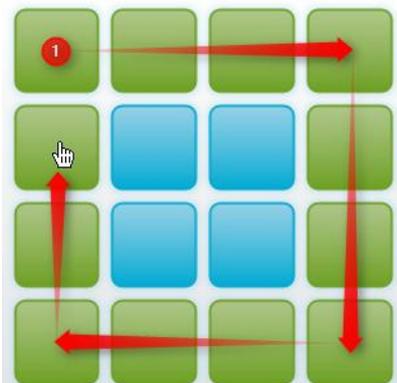
Administrator:

Maintenance:

Operator:

In addition there is a logout button which logs out the currently logged in user and opens the login screen. The login screen is a modal dialog which dims the background.

7.1 Equipment screen

The equipment screen offers the alarm line with the same functionality as on the start page. filtered for the piece of equipment.

In addition, there is the Workspace concept with freely positionable windows, which are stored in a Dock when they are not used. The Workspace spans several screens to which you can change via Swipe gesture, Tab navigation or navigation button. In the lower area there is a activation area for two-hand operation and a home button. At the right top edge there is an operable display for the Workspace.

DOCK

Icons can be dragged from the   Dock to the   Workspace where they are then displayed as a   Faceplate in a defined base size. If a Faceplate is placed on the Workspace , its Icon is displayed as deactivated. If a Faceplate is closed, its Icon is activated again.   Tap & hold on a deactivated   Icon pinpoints an open Faceplate and jumps to the Workspace used by it.

WORKSPACE

You can place an scale any number of   Workspaces on each of the four personalizable Faceplates Faceplates

- ▶ Faceplate Move to the vertical screen edge: After a delay of 2 seconds, a change to the next Workspace   is carried out and the   Faceplate can be positioned freely.

- ▶ Faceplate Jump to the vertical screen edge (Swipe gesture): The   Faceplate is moved to the next Workspace, the current   Workspace remains open.

- ▶ Faceplate Close/remove: Faceplate Move to the dock or down via a   Swipe gesture.
  As an option, each   Faceplate can be closed via the **x** button, which is located at the top right corner.

The   Faceplates called up, their position, size, etc. are saved in the user profile. A   Faceplate can be resized (larger/smaller) via a   Zoom/Pinch gesture. Each   Faceplate   can also be moved. A selected Faceplate is moved to the foreground via Z-Order-Manipulation but always remains behind the alarm line. At the next login the position and size data of the individual Faceplates   are read and they are positioned accordingly.

TWO-HAND OPERATION

In the bottom left corner there is an activation area for two-hand operation. If a locked element is actuated, the activation area flashes and a locked element can be unlocked via this area.

Possibilities for two-hand operation:

▶   Button: for example home

▶   Input set value: Keyboard is called up and set value can be entered,

▶   Jog operation

A consideration of the activation order (activation before action) is engineered in the demo project. At jog operation (Faceplate   operation) the active activation is constantly checked.

CIRCLE MENU

The circle menu was implemented for faster navigation between   Workspaces. It is activated using Tap&Hold anywhere on the   Workspace and is displayed around the finger touching the screen. The workspace is selected by   Draggen   the finger in one of the areas. The switch is carried out when the finger leaves the screen.   The action can be canceled by Draggen outside or inside the menu area.

HOME BUTTON

There is a   Home button in the right bottom corner. With the help of the home button you can switch to the start screen. The Home button can only be activated with two-hand operation.

7.2   VSTA Code

CLASS DESCRIPTION

MULTITOUCHMANAGEMENT

Complete handling of the whole Multi-Touch application. At creating the `MultitouchManagement` class, the classes `LoginWindow`, `NavigationsWindow` and `WindowManagement` are instanced.

LOGINWINDOW

In this class the important component of the user login and the password pattern recognition are included.

NAVIGATIONWINDOW

Handles the "`Icon`" faceplate positioning screen and manages the whole opening process of the Faceplates which are called up.

WINDOWMANAGEMENT

Is responsible for processing all touch events of all faceplates (move, scale, etc.). In addition, this class takes care of saving and reading all needed Faceplate information in the Runtime.

WINDOWPROPERTIES

For each   Faceplate, an independent instance is instanced, which provides all necessary data of the Faceplates . In terms of data retention, all instances are saved in an XML file when Runtime is closed and can therefore provide the last valid settings of the Faceplates at the start of Runtime.

7.3   Basic information on Multi-Touch with Windows 7

EVENTS

If you activate project setting `Multi-Touch active`, you get the events for **`TouchManipulationStartEvent`**, **`TouchManipulationDeltaEvent`**, and **`TouchManipulationCompleteEvent`** for corresponding event handler declaration in the Runtime. Via method **`SetupTouchInertia`** you can define the inertial parameter for each screen. The following display shows a schematic process of the fired events:
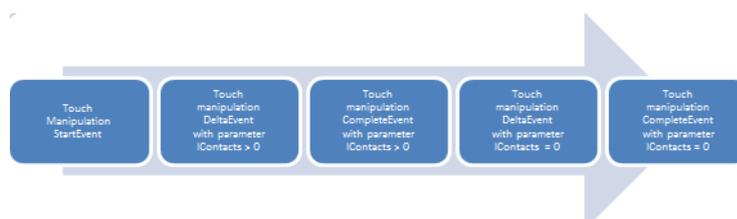
Generally a **TouchManipulationStartEvent** is fired first. As long as you execute the gesture, **TouchManipulationDeltaEvents** and at the end of the gesture a concluding **TouchManipulationCompleteEvent** is fired. The handed over parameter `IContacts` gives back the number of the fingers currently on the screen.

After finishing the touch gesture the inertia values are calculated by the inertia processor via the values handed over by method **SetupTouchInertia** and finished via **TouchManipulationDeltaEvents** and the calculated inertia values with a singular **TouchManipulationCompleteEvent**.

As no finger is on the screen during the calculation of the inertia values, the handed over IContacts parameter is `0`. The number of the **TouchManipulationDeltaEvents** needed by the inertia processor depends on the handed over parameters by method **SetupTouchInertia**. Depending on the inertia the inertia processor needs more or less events to finish the gesture.

If during the firing of the calculated inertia events another gesture is started, no additional **TouchManipulationDeltaEvents** come from the old gesture. After a concluding **TouchManipulationCompleteEvents**, the events for the new gesture are started immediately via a **TouchManipulationStartEvents**.

In addition events **TouchEvent**, **ElementMouseOver**, **ElementLeftButtonDown** and **ElementLeftButtonUp** are fired at a gesture.

NATIVE MULTI-TOUCH IN THE WORLDVIEW

To implement zooming and scrolling via VBA/VSTA Events, property `Multi-Touch for zoom and scroll` must not be active. If you activate this property, zenon take care about the zoom and scroll gestures in the engineered worldview.   For more information see chapter: Navigation with Multi-Touch in the worldview (on page 24).

7.4   FAQs

Frequently asked questions and practical answers.

Note: The properties `Name for object list` and `Help chapter` can be used in zenon as freely definable properties.

HOW DO I ADD A NEW FACEPLATE?

The example project consists of 13 Faceplates, which are displayed as Icons in the **Navigation_Bottom** navigation screen. To generate a context between the individual icons and the Faceplates to be used, you

must adhere to the name convention. The name of the respective Icons must be a perfect match with the corresponding screen and its frame. A connection between the screens can only be accomplished when this chain (Icon - frame - screen) is observed.

To call up the respective screens, you also must engineer a screen switch function. The name of the screen switch function consists of the prefix "**scr**" and the name of the Faceplates. If this name convention is observed, the complete handling is then managed by the Multi-Touch Management.

HOW TO CREATE FEEDBACK IF THE PRESSED BUTTON OR THE VALUE DISPLAY ELEMENT NEEDS TWO-HAND OPERATION?

To ensure two-hand operation, both pressure points must be provided each on an own screen and frame. In the example project, the screen is called Enabler.

At engineering the interlocking must be linked with **enableArea** for two-hand operable elements. This engineering makes sure that feedback is automatically generated provided the Enabler (two-hand operation) is not pressed. This element is only operable if the Enabler is pressed.

HOW TO CREATE A JOG MODE BUTTON?

To create a button with a jog operation functionality:

- ▶ make sure that the name of the button contains the part name "HOLD"

- ▶ navigate to the Runtime node in the element properties

- ▶ enter a reference according to the following pattern in property Name for object list:
  Variable name|value change
  (You can find an example in the Faceplate_Operation screen - the ^^^ button.)

HOW TO CREATE A BUTTON FOR CHANGING THE WORKSPACE?

For this you can use prefab symbol **Tab**. The variable **demoCurrentWorkspace** depicts the value of the currently active Workspace in the whole project. The following settings manipulate the variable and newly adjust all opened screens:

Group **tab_inactive:**

The Help chapter property must contain the WorkspaceSwitch entry.

property Name for object list must contain an entry according to the following pattern:
demoCurrentWorkspace|PAGE (for example: **demoCurrentWorkspace|3**)

HOW TO CREATE A NEW USER?

In screen `Login` you must draw a user-specific pattern on the 16 possible square for logging in a user. As soon as you touch the first square, the password input is triggered. It is closed as soon as the finger leaves the screen. Immediately after the pattern is entered, the password is verified. The login is carried out automatically after positive acknowledgment.

During the input of the password pattern a typographic password is created in the background which can be compared with the engineered project users via a function. The first square is interpreted as `'A'`, the second as `'B'`, etc. and put together to a coherent password via line-dependent hovering of the squares.

To add a new user:

- ▶ create a new user in the user administration

- ▶ create a personal password

- ▶ make sure that the user can be selected in screen login and engineer the same properties as for the already engineered buttons of the user selection

HOW TO CALL UP THE CIRCLE MENU?

The circle menu is opened.

- ▶ around a finger which touches the workspace
  and

- ▶ does not move for more than `800 ms`

In the circle menu, you have the possibility to switch between the Workspaces to the desired Workspace by dragging your finger.

HOW TO FILTER THE FREELY PULL-DOWN AML?

The AML always stays called up and the equipment-based filtering is carried out by a simple filter possibility of the screen switch function to screen `AML` and `AML-collapsed`.

HOW TO CALL UP A PIECE OF EQUIPMENT?

Items of equipment are displayed via a fixed order of processes such as customizing the filter, screen switch function, calling up equipment-related navigation areas, positioning the equipment-dependent

Faceplates, etc. In this example project, the order of the functions which are to be called up is engineered in a zenon script, which is called up at triggering the equipment `Icons` in the start screen.

WHERE IS THE INFORMATION OF THE FACEPLATES SAVED?

Size, position, workspace, visibility, and scaling factor are written in a configuration file. These are located in a subfolder of folder
`%HOMEPATH%\AppData\Local\Ing._Punzenberger_COPA-DA\`.

8. Navigation with Multi-Touch in the worldview

Multi-Touch gestures for zooming and scrolling are suitable for navigation on touch panels in the worldview. For this a screen of type `Worldview overview` is not necessary. The navigation can be implemented with:

8.1 Navigation under Windows 7

To be able to use Multi-Touch gestures under Windows 7 to navigate in the worldview, you must:

- ▶ activate them via property `Multi-Touch for zoom and scroll`
- ▶ or implement them via VBA/VSTA

ZOOM AND SCROLL VIA PROPERTY MULTI-TOUCH FOR ZOOM AND SCROLL

To use Multi-Touch without VBA/VSTA:

1. In the project properties in the `Interaction` note for the `Recognition` property, activate `Windows 7` or `Windows 8 Multi-Touch active`

2. deactivate property `Size from frame` in node `Size` at the properties of the screen

3. activate property `Multi-Touch for zoom and scroll` in node `Interaction` at the properties of the screen

With this you can scroll and zoom in the screen with touch operation using Multi-Touch gestures. With this VBA/VSTA for zooming and scrolling is deactivated.

ZOOM AND SCROLL VIA VBA/VSTA

To implement zooming and scrolling via VBA/VSTA Events, property `Multi-Touch for zoom and scroll` must not be active.

The following is available in the `DynPicture`:

▸ Property

`int ZoomLevel`: Displays the current zoom level in the worldview (valid value only in the Runtime and for a worldview).

▸ Method

`SetZoomAndPos(float ZoomX, float ZoomY, int ZoomLevel, int CursorX, int CursorY, int PosX, int PosY, int PosMode)`:

`ZoomX ->` New zoom factor X direction; if not used, set to 0

`ZoomY ->` New zoom factor Y direction; if not used, set to 0

`ZoomLevel ->` Zoom level, if not used, set to -1

`CursorX ->` Cursorposition X

`CursorY ->` Cursorposition Y

`PosX ->` New position X (see PosMode)

`PosY ->` New position Y (see PosMode)

`PosMode ->` Coordinates in Pos

-1 = PosX, PosY are ignored

0 = center point , original coordinates

1 = center point, zoomed coordinates

2 = left top, original coordinates

3 = left top, zoomed coordinates

4 = zoomed coordinates of the cursor from the top left

The position of the window is changed in such a way that after the zooming the mouse cursor is still over the position of the screen

Attention: `ZoomX`, `ZoomY` and `ZoomLevel` can never be used simultaneously. Either you enter a `ZoomLevel` or a zoom factor for x and y axis.

8.2   Navigation under Windows 8

To navigate in a worldview with Multi-Touch under Windows 8:

1. Deactivate the `Size from frame` property for the screen in the `Size` group

2. navigate to the `Interaction` group in the screen properties.

3. Configure the properties for   `Zoomen` and `Verschieben`.

For the move gesture, you can define the direction - horizontal, vertical or both. To do this, use the `Horizontal verschieben` and `Vertikal verschieben` properties.