

SIEMENS

SIMATIC HMI

WinCC V6 VBS Reference

Printout of the Online Help

Edition April 2003
A5E00243392

Contents

1	VBS Reference	1
1.1	Objects and Lists.....	2
1.1.1	ActiveScreen Object.....	3
1.1.2	ActiveScreenItem Object	3
1.1.3	HMIRuntime Object	3
1.1.4	Item Object	4
1.1.5	Layer Object	5
1.1.6	Layers Object (List)	6
1.1.7	Parent Object.....	6
1.1.8	ScreenItem Object.....	8
1.1.9	ScreenItems Object (List).....	10
1.1.10	Screen Object	11
1.1.11	Screens Object (List).....	13
1.1.12	Tag Object.....	14
1.1.13	Tags Object (List).....	16
1.2	Object Type of the ScreenItem Object	17
1.2.1	Standard objects	18
1.2.1.1	Ellipse	18
1.2.1.2	Ellipse Arc	18
1.2.1.3	Ellipse Segment	19
1.2.1.4	Circle.....	20
1.2.1.5	Circular Arc	20
1.2.1.6	Pie Segment	21
1.2.1.7	Line	22
1.2.1.8	Polygon	22
1.2.1.9	Polyline	23
1.2.1.10	Rectangle	24
1.2.1.11	Rounded Rectangle.....	25
1.2.1.12	Static Text	27
1.2.1.13	Connector.....	28
1.2.2	Smart Objects	28
1.2.2.1	3D-Bar.....	29
1.2.2.2	Application Window	30
1.2.2.3	Bar.....	31
1.2.2.4	Picture Window.....	32
1.2.2.5	Control	33
1.2.2.6	I/O Field.....	35
1.2.2.7	Graphic Object.....	35
1.2.2.8	OLE Object.....	36
1.2.2.9	Text List.....	38
1.2.2.10	Status Display	39
1.2.3	Windows Objects	40
1.2.3.1	Button	40
1.2.3.2	Check Box	41
1.2.3.3	Option Group	42
1.2.3.4	Round Button.....	42
1.2.3.5	Slider.....	43
1.2.4	Controls.....	45
1.2.4.1	WinCC Alarm Control	46
1.2.4.2	WinCC Digital Analog Clock.....	47

1.2.4.3	WinCC DXF Control	48
1.2.4.4	WinCC Function Trend Control.....	48
1.2.4.5	WinCC Gauge Control.....	48
1.2.4.6	WinCC Online Table Control	49
1.2.4.7	WinCC Online Trend Control.....	49
1.2.4.8	WinCC Push Button Control.....	51
1.2.4.9	WinCC Slider Control	53
1.2.4.10	HMI Symbol Library 1.3	54
1.2.5	Customized object.....	55
1.2.6	Group object.....	55
1.3	Characteristics.....	55
1.3.1	A.....	57
1.3.1.1	AccessPath Property	57
1.3.1.2	Activate Property.....	57
1.3.1.3	ActiveScreen Property	57
1.3.1.4	ActiveScreenItem Property	57
1.3.1.5	Actualize Property	58
1.3.1.6	ActualPointLeft Property	58
1.3.1.7	ActualPointTop Property	58
1.3.1.8	AdaptBorder Property.....	58
1.3.1.9	AdaptPicture Property.....	59
1.3.1.10	AdaptSize Property	59
1.3.1.11	AlarmHigh Property	59
1.3.1.12	AlarmLow Property.....	59
1.3.1.13	Alignment Property.....	60
1.3.1.14	AlignmentLeft Property	60
1.3.1.15	AlignmentTop Property	60
1.3.1.16	AllowPersistence Property	60
1.3.1.17	AllServer Property	60
1.3.1.18	Analog Property	61
1.3.1.19	AngleAlpha Property.....	61
1.3.1.20	AngleBeta Property	61
1.3.1.21	AngleMax Property.....	61
1.3.1.22	AngleMin Property.....	61
1.3.1.23	Application Property	61
1.3.1.24	Archive Property	62
1.3.1.25	Assignments Property.....	62
1.3.1.26	AssumeOnExit Property.....	62
1.3.1.27	AssumeOnFull Property.....	62
1.3.1.28	Autorange Property	62
1.3.1.29	AutorangeX Property	63
1.3.1.30	AutorangeY Property	63
1.3.1.31	AutoScroll Property	63
1.3.1.32	AutoSize Property	63
1.3.1.33	Average Property	63
1.3.1.34	Axe Property.....	64
1.3.1.35	AxisSection Property	64
1.3.2	B.....	64
1.3.2.1	BackBorderWidth Property.....	64
1.3.2.2	BackColor Property	64
1.3.2.3	BackColor2 Property	65
1.3.2.4	BackColor3 Property	65
1.3.2.5	BackColorBottom Property.....	65
1.3.2.6	BackColorTop Property	65
1.3.2.7	BackFlashColorOff Property.....	66

1.3.2.8	BackFlashColorOn Property.....	66
1.3.2.9	Background Property.....	66
1.3.2.10	BackgroundPicture Property	66
1.3.2.11	BackStyle Property.....	66
1.3.2.12	BarBackColor Property.....	67
1.3.2.13	BarDepth Property.....	67
1.3.2.14	BarFillColor Property	67
1.3.2.15	BarHeight Property.....	67
1.3.2.16	BarWidth Property.....	67
1.3.2.17	BasePicReferenced Property	67
1.3.2.18	BasePicTransColor Property.....	67
1.3.2.19	BasePicture Property.....	67
1.3.2.20	BasePicUseTransColor Property	67
1.3.2.21	BaseScreenName Property.....	68
1.3.2.22	BaseY Property.....	69
1.3.2.23	BaseX Property.....	69
1.3.2.24	BeginTime Property.....	69
1.3.2.25	BeginValue Property.....	69
1.3.2.26	BeginX Property.....	69
1.3.2.27	BeginY Property.....	69
1.3.2.28	BevelColorDown Property.....	70
1.3.2.29	BevelColorUp Property	70
1.3.2.30	BevelInner Property.....	70
1.3.2.31	BevelOuter Property	70
1.3.2.32	BevelWidth Property.....	71
1.3.2.33	BitNumber Property.....	71
1.3.2.34	BlinkColor Property	71
1.3.2.35	BlinkMode Property	71
1.3.2.36	BlinkSpeed Property.....	71
1.3.2.37	BorderBackColor Property	71
1.3.2.38	BorderColor Property.....	71
1.3.2.39	BorderColorBottom Property	71
1.3.2.40	BorderColorTop Property	72
1.3.2.41	BorderEndStyle Property	72
1.3.2.42	BorderFlashColorOff Property	72
1.3.2.43	BorderFlashColorOn Property	72
1.3.2.44	BorderStyle Property	72
1.3.2.45	BorderWidth Property	73
1.3.2.46	BottomConnectedConnectionPointIndex Property	73
1.3.2.47	BottomConnectedObjectName Property	73
1.3.2.48	BoxAlignment Property	73
1.3.2.49	BoxCount Property	73
1.3.2.50	BoxType Property	73
1.3.2.51	ButtonColor Property.....	74
1.3.2.52	ButtonCommand Property.....	74
1.3.2.53	Button1Width Property.....	75
1.3.2.54	Button2Width Property.....	75
1.3.2.55	Button3Width Property.....	75
1.3.2.56	Button4Width Property.....	75
1.3.3	C.....	75
1.3.3.1	Caption Property	75
1.3.3.2	CaptionColor Property	76
1.3.3.3	CaptionFont Property	76
1.3.3.4	CaptionOffset Property	76
1.3.3.5	CaptionText Property.....	76

1.3.3.6	CellCut Property.....	76
1.3.3.7	CenterColor Property.....	76
1.3.3.8	CenterScale Property.....	77
1.3.3.9	CheckAlarmHigh Property.....	77
1.3.3.10	CheckAlarmLow Property.....	77
1.3.3.11	CheckLimitHigh4 Property.....	77
1.3.3.12	CheckLimitHigh5 Property.....	77
1.3.3.13	CheckLimitLow4 Property.....	78
1.3.3.14	CheckLimitLow5 Property.....	78
1.3.3.15	CheckToleranceHigh Property.....	78
1.3.3.16	CheckToleranceLow Property.....	78
1.3.3.17	CheckWarningHigh Property.....	78
1.3.3.18	CheckWarningLow Property.....	79
1.3.3.19	ClearOnError Property.....	79
1.3.3.20	ClearOnNew Property.....	79
1.3.3.21	Closeable Property.....	79
1.3.3.22	CloseButton Property.....	79
1.3.3.23	CoarseGrid Property.....	79
1.3.3.24	CoarseGridX Property.....	79
1.3.3.25	CoarseGridY Property.....	80
1.3.3.26	CoarseGridValue Property.....	80
1.3.3.27	CoarseGridValueX Property.....	80
1.3.3.28	CoarseGridValueY Property.....	80
1.3.3.29	CollectValue Property.....	80
1.3.3.30	ColMove Property.....	80
1.3.3.31	Color Property.....	80
1.3.3.32	ColorAlarmHigh Property.....	81
1.3.3.33	ColorAlarmLow Property.....	81
1.3.3.34	ColorBottom Property.....	81
1.3.3.35	ColorChangeType Property.....	81
1.3.3.36	ColorLimitHigh4 Property.....	81
1.3.3.37	ColorLimitHigh5 Property.....	81
1.3.3.38	ColorLimitLow4 Property.....	82
1.3.3.39	ColorLimitLow5 Property.....	82
1.3.3.40	ColorToleranceHigh Property.....	82
1.3.3.41	ColorToleranceLow Property.....	82
1.3.3.42	ColorTop Property.....	82
1.3.3.43	ColorWarningHigh Property.....	82
1.3.3.44	ColorWarningLow Property.....	83
1.3.3.45	ColTitle Property.....	83
1.3.3.46	Command Property.....	83
1.3.3.47	ColWidth Property.....	83
1.3.3.48	CommonTime Property.....	83
1.3.3.49	CommonX Property.....	83
1.3.3.50	CommonY Property.....	83
1.3.3.51	ContinousChange Property.....	84
1.3.3.52	CurrentContext Property.....	84
1.3.3.53	Cursor Property.....	84
1.3.3.54	CursorControl Property.....	84
1.3.3.55	CurveForm Property.....	85
1.3.4	D.....	85
1.3.4.1	DangerColor Property.....	85
1.3.4.2	Danger Property.....	85
1.3.4.3	DataFormat Property.....	85
1.3.4.4	DataIndex Property.....	86

1.3.4.5	DataX Property	86
1.3.4.6	DataY Property	86
1.3.4.7	DeleteData Property	86
1.3.4.8	Delta Property	86
1.3.4.9	DesiredCurveColor Property	86
1.3.4.10	DesiredCurveCurveForm Property	86
1.3.4.11	DesiredCurveSourceNumberOfUAValues Property	87
1.3.4.12	DesiredCurveSourceUAArchive Property	87
1.3.4.13	DesiredCurveSourceUAArchiveStartID Property	87
1.3.4.14	DesiredCurveSourceUAColumnX Property	87
1.3.4.15	DesiredCurveSourceUAColumnY Property	87
1.3.4.16	DesiredCurveVisible Property	88
1.3.4.17	Direction Property	88
1.3.4.18	DXFFilename Property	88
1.3.4.19	E	88
1.3.5	E	89
1.3.5.1	Edit Property	89
1.3.5.2	Editable Property	89
1.3.5.3	EditAtOnce Property	89
1.3.5.4	Enabled Property	89
1.3.5.5	EndAngle Property	90
1.3.5.6	EndTime Property	90
1.3.5.7	EndValue Property	90
1.3.5.8	EndX Property	90
1.3.5.9	EndY Property	90
1.3.5.10	Error Property	90
1.3.5.11	Exponent Property	91
1.3.5.12	ExtendedOperation Property	91
1.3.6	F	91
1.3.6.1	FillColor Property	91
1.3.6.2	Filling Property	92
1.3.6.3	FillingIndex Property	92
1.3.6.4	FillStyle Property	92
1.3.6.5	FillStyle2 Property	93
1.3.6.6	FineGrid Property	93
1.3.6.7	FineGridValue Property	93
1.3.6.8	FineGridValueX Property	93
1.3.6.9	FineGridValueY Property	94
1.3.6.10	FineGridX Property	94
1.3.6.11	FineGridY Property	94
1.3.6.12	FlashBackColor Property	94
1.3.6.13	FlashBorderColor Property	94
1.3.6.14	FlashFlashPicture Property	94
1.3.6.15	FlashForeColor Property	94
1.3.6.16	FlashPicReferenced Property	94
1.3.6.17	FlashPicTransColor Property	95
1.3.6.18	FlashPicture Property	95
1.3.6.19	FlashPicUseTransColor Property	95
1.3.6.20	FlashRate Property	95
1.3.6.21	FlashRateBackColor Property	95
1.3.6.22	FlashRateBorderColor Property	96
1.3.6.23	FlashRateFlashPic Property	96
1.3.6.24	FlashRateForeColor Property	96
1.3.6.25	Flip Property	96
1.3.6.26	FocusColor Property	96

1.3.6.27	FocusRect Property	97
1.3.6.28	FocusWidth Property	97
1.3.6.29	Font Property	97
1.3.6.30	FontBold Property	97
1.3.6.31	FontItalic Property	98
1.3.6.32	FontName Property	98
1.3.6.33	FontPosition Property	98
1.3.6.34	FontSize Property	98
1.3.6.35	FontStrikeThru Property	98
1.3.6.36	FontUnderline Property	98
1.3.6.37	ForeColor Property	98
1.3.6.38	ForeFlashColorOff Property	99
1.3.6.39	ForeFlashColorOn Property	99
1.3.6.40	FrameColor Property	99
1.3.6.41	FrameColorDown Property	99
1.3.6.42	FrameColorUp Property	99
1.3.6.43	FramePicture Property	99
1.3.6.44	FrameScale Property	99
1.3.6.45	FrameWidth Property	100
1.3.6.46	FreezeProviderConnections Property	100
1.3.7	G	100
1.3.7.1	GraphDirection Property	100
1.3.7.2	GridLineHorz Property	100
1.3.7.3	GridLines Property	100
1.3.7.4	GridLineValue Property	101
1.3.7.5	GridlinesValueX Property	101
1.3.7.6	GridlinesValueY Property	101
1.3.7.7	GridLineVert Property	101
1.3.7.8	GridlinesX Property	101
1.3.7.9	GridlinesY Property	101
1.3.8	H	101
1.3.8.1	andFillColor Property	101
1.3.8.2	Handtype Property	101
1.3.8.3	Height Property	101
1.3.8.4	HiddenInput Property	102
1.3.8.5	Hotkey Property	103
1.3.8.6	HourNeedleHeight Property	103
1.3.8.7	HourNeedleWidth Property	103
1.3.8.8	Hysteresis Property	103
1.3.8.9	HysteresisRange Property	103
1.3.9	Index Property	104
1.3.9.1	InnerBevelOffset Property	104
1.3.9.2	InnerBevelStyle Property	104
1.3.9.3	InnerBevelWidth Property	104
1.3.9.4	InsertData Property	105
1.3.9.5	ItemBorderBackColor Property	105
1.3.9.6	ItemBorderColor Property	105
1.3.9.7	ItemBorderStyle Property	105
1.3.9.8	ItemBorderWidth Property	105
1.3.9.9	ItemVisible Property	105
1.3.10	L	106
1.3.10.1	Label Property	106
1.3.10.2	LabelColor Property	106
1.3.10.3	LabelX Property	106
1.3.10.4	LabelY Property	106

1.3.10.5	LanguageSwitch Property	106
1.3.10.6	Language Property	106
1.3.10.7	LastError Property	106
1.3.10.8	Layer Property	107
1.3.10.9	Layer00Checked Property	108
1.3.10.10	Layer01Checked Property	108
1.3.10.11	Layer02Checked Property	108
1.3.10.12	Layer03Checked Property	108
1.3.10.13	Layer04Checked Property	109
1.3.10.14	Layer05Checked Property	109
1.3.10.15	Layer06Checked Property	109
1.3.10.16	Layer07Checked Property	109
1.3.10.17	Layer08Checked Property	109
1.3.10.18	Layer09Checked Property	110
1.3.10.19	Layer10Checked Property	110
1.3.10.20	Layer00Color Property.....	110
1.3.10.21	Layer01Color Property.....	110
1.3.10.22	Layer02Color Property.....	110
1.3.10.23	Layer03Color Property.....	110
1.3.10.24	Layer04Color Property.....	111
1.3.10.25	Layer05Color Property.....	111
1.3.10.26	Layer06Color Property.....	111
1.3.10.27	Layer07Color Property.....	111
1.3.10.28	Layer08Color Property.....	111
1.3.10.29	Layer09Color Property.....	111
1.3.10.30	Layer10Color Property.....	112
1.3.10.31	Layer00Value Property	112
1.3.10.32	Layer01Value Property	112
1.3.10.33	Layer02Value Property	112
1.3.10.34	Layer03Value Property	112
1.3.10.35	Layer04Value Property	112
1.3.10.36	Layer05Value Property	113
1.3.10.37	Layer06Value Property	113
1.3.10.38	Layer07Value Property	113
1.3.10.39	Layer08Value Property	113
1.3.10.40	Layer09Value Property	113
1.3.10.41	Layer10Value Property	113
1.3.10.42	Left Property	114
1.3.10.43	LeftComma Property	114
1.3.10.44	LightEffect Property	114
1.3.10.45	LimitHigh4 Property	114
1.3.10.46	LimitHigh5 Property	115
1.3.10.47	LimitLow4 Property	115
1.3.10.48	LimitLow5 Property	115
1.3.10.49	LimitMax Property	115
1.3.10.50	LimitMin Property	115
1.3.10.51	LineFont Property.....	115
1.3.10.52	LineHeight Property.....	115
1.3.10.53	LineTitle Property	115
1.3.10.54	ListType Property	116
1.3.10.55	LoadDataImmediately Property	116
1.3.10.56	LocaleID Property	116
1.3.10.57	LockBackColor Property	116
1.3.10.58	LockStatus Property	117
1.3.10.59	LockText Property	117

1.3.10.60	LockTextColor Property	117
1.3.10.61	LongStrokesBold Property	117
1.3.10.62	LongStrokesOnly Property	117
1.3.10.63	LongStrokesSize Property	117
1.3.10.64	LongStrokesTextEach Property	117
1.3.10.65	LowerLimit Property	118
1.3.10.66	LowerLimitColor Property	118
1.3.10.67	LowerLimitValue Property	118
1.3.11	M	118
1.3.11.1	Marker Property	118
1.3.11.2	Max Property	119
1.3.11.3	MaximizeButton Property	119
1.3.11.4	MCGUBackColorOff Property	119
1.3.11.5	MCGUBackColorOn Property	119
1.3.11.6	MCGUBackFlash Property	119
1.3.11.7	MCGUTextColorOff Property	119
1.3.11.8	MCGUTextColorOn Property	119
1.3.11.9	MCGUTextFlash Property	120
1.3.11.10	MCKOBackColorOff Property	120
1.3.11.11	MCKOBackColorOn Property	120
1.3.11.12	MCKOBackFlash Property	120
1.3.11.13	MCKOTextColorOff Property	120
1.3.11.14	MCKOTextColorOn Property	120
1.3.11.15	MCKOTextFlash Property	120
1.3.11.16	MCKQBackColorOff Property	121
1.3.11.17	MCKQBackColorOn Property	121
1.3.11.18	MCKQBackFlash Property	121
1.3.11.19	MCKQTextColorOff Property	121
1.3.11.20	MCKQTextColorOn Property	121
1.3.11.21	MCKQTextFlash Property	121
1.3.11.22	MCText Property	121
1.3.11.23	MeasurePoints Property	122
1.3.11.24	MessageClass Property	122
1.3.11.25	Min Property	122
1.3.11.26	MinuteNeedleHeight Property	122
1.3.11.27	MinuteNeedleWidth Property	122
1.3.11.28	Moveable Property	122
1.3.11.29	MsgCtrlFlags Property	123
1.3.11.30	MsgFilterSQL Property	123
1.3.12	N	123
1.3.13	Name Property	123
1.3.13.1	NeedleColor Property	124
1.3.13.2	NormalColor Property	124
1.3.13.3	NumberLines Property	124
1.3.13.4	NumItems Property	124
1.3.14	O	124
1.3.14.1	Object Property	124
1.3.14.2	ObjectName Property	125
1.3.14.3	OffsetLeft Property	125
1.3.14.4	OffsetTop Property	125
1.3.14.5	Online Property	126
1.3.14.6	OnTop Property	126
1.3.14.7	OperationMessage Property	126
1.3.14.8	OperationReport Property	126
1.3.14.9	Orientation Property	127

1.3.14.10	OuterBevelStyle Property.....	127
1.3.14.11	OuterBevelWidth Property	127
1.3.14.12	Outline Property	127
1.3.14.13	OutputFormat Property	127
1.3.14.14	OutputValue Property	127
1.3.15	P-Q	128
1.3.15.1	arent Property	128
1.3.15.2	PasswordLevel Property	128
1.3.15.3	PersistentRT Property	128
1.3.15.4	PersistentRTCS Property	128
1.3.15.5	PersistentRTCSPermission Property	128
1.3.15.6	PersistentRTPermission Property	129
1.3.15.7	PicDeactReferenced Property	129
1.3.15.8	PicDeactTransparent Property	129
1.3.15.9	PicDeactUseTransColor Property	129
1.3.15.10	PicDownReferenced Property	129
1.3.15.11	PicDownTransparent Property	129
1.3.15.12	PicDownUseTransColor Property	130
1.3.15.13	PicReferenced Property	130
1.3.15.14	PicTransColor Property	130
1.3.15.15	Picture Property	130
1.3.15.16	PictureBack Property	130
1.3.15.17	PictureDeactivated Property.....	130
1.3.15.18	PictureDown Property.....	130
1.3.15.19	PictureUp Property	130
1.3.15.20	PictureName Property	130
1.3.15.21	PictureSelected Property	131
1.3.15.22	PictureThumb Property.....	131
1.3.15.23	PictureUnselected Property.....	131
1.3.15.24	PicUpReferenced Property	131
1.3.15.25	PicUpTransparent Property.....	131
1.3.15.26	PicUpUseTransColor Property	131
1.3.15.27	PicUseTransColor Property.....	131
1.3.15.28	Position Property.....	131
1.3.15.29	Precisions Property	131
1.3.15.30	PrecisionX Property.....	132
1.3.15.31	PrecisionY Property.....	132
1.3.15.32	PredefinedAngles Property	133
1.3.15.33	Pressed Property	133
1.3.15.34	PrintJob Property	133
1.3.15.35	Process Property.....	133
1.3.15.36	ProjectPath Property	133
1.3.15.37	ProviderClsid Property.....	133
1.3.15.38	ProviderType Property.....	134
1.3.15.39	PointCount Property	134
1.3.15.40	QualityCode Property	134
1.3.16	R.....	135
1.3.16.1	Radius Property	135
1.3.16.2	RadiusHeight Property.....	135
1.3.16.3	RadiusWidth Property.....	135
1.3.16.4	RangeMax Property	135
1.3.16.5	RangeMin Property	135
1.3.16.6	Rectangular Property.....	135
1.3.16.7	ReferenceRotationLeft Property	136
1.3.16.8	ReferenceRotationTop Property	136

1.3.16.9	RelayCurves Property.....	136
1.3.16.10	Relevant Property	136
1.3.16.11	Replacement Property	136
1.3.16.12	ReplacementColor Property	136
1.3.16.13	RightComma Property	137
1.3.16.14	Rotation Property	137
1.3.16.15	RotationAngle Property.....	137
1.3.16.16	RoundCornerHeight Property	137
1.3.16.17	RoundCornerWidth Property	137
1.3.16.18	RulerPrecisions Property	137
1.3.16.19	RulerPrecisionX Property	138
1.3.16.20	RulerPrecisionY Property	138
1.3.17	S.....	138
1.3.17.1	SameSize Property	138
1.3.17.2	ScaleColor Property	138
1.3.17.3	ScaleTicks Property	138
1.3.17.4	Scaling Property.....	138
1.3.17.5	ScalingType Property	139
1.3.17.6	ScalingTypeX Property	139
1.3.17.7	ScalingTypeY Property	139
1.3.17.8	Screen Property	139
1.3.17.9	ScreenName Property	140
1.3.17.10	Screens Property	140
1.3.17.11	ScreenItems Property.....	140
1.3.17.12	Scrollable Property	140
1.3.17.13	ScrollBars Property	141
1.3.17.14	SecondNeedleHeight Property	141
1.3.17.15	SecondNeedleWidth Property	141
1.3.17.16	SelBGColor Property	141
1.3.17.17	SelectionMode Property.....	142
1.3.17.18	SelTextColor Property	142
1.3.17.19	ServerData Property	142
1.3.17.20	ServerPrefix Property	142
1.3.17.21	ServerNames Property	142
1.3.17.22	ShowBar Property	143
1.3.17.23	ShowDanger Property	143
1.3.17.24	ShowDecimalPoint Property.....	143
1.3.17.25	ShowNormal Property.....	143
1.3.17.26	ShowPeak Property.....	143
1.3.17.27	ShowPosition Property	144
1.3.17.28	ShowRulerImmediately Property	144
1.3.17.29	ShowThumb Property	144
1.3.17.30	ShowValuesExponentialX Property	144
1.3.17.31	ShowValuesExponentialY Property	144
1.3.17.32	ShowWarning Property.....	144
1.3.17.33	SignificantMask Property	145
1.3.17.34	Sizeable Property.....	145
1.3.17.35	SmallChange Property.....	145
1.3.17.36	SourceBeginTime Property	145
1.3.17.37	SourceEndTime Property	145
1.3.17.38	SourceNumberOfUAValues Property	146
1.3.17.39	SourceNumberOfValues Property	146
1.3.17.40	SourceTagNameX Property	146
1.3.17.41	SourceTagNameY Property	146
1.3.17.42	SourceTagProviderDataX Property.....	147

1.3.17.43	SourceTagProviderDataY Property	147
1.3.17.44	SourceTimeRange Property	147
1.3.17.45	SourceUAArchive Property	147
1.3.17.46	SourceUAArchiveStartID Property	148
1.3.17.47	SourceUAColumnX Property	148
1.3.17.48	SourceUAColumnY Property	148
1.3.17.49	SquareExtent Property	148
1.3.17.50	StartAngle Property	148
1.3.17.51	Statusbar Property	148
1.3.17.52	StatusbarPanels Property	148
1.3.17.53	Stretch Property	149
1.3.18	T	149
1.3.18.1	agName Property	149
1.3.18.2	TagPrefix Property	150
1.3.18.3	Tags Property	150
1.3.18.4	TagProviderClsid Property	150
1.3.18.5	Template Property	150
1.3.18.6	Text Property	151
1.3.18.7	ThumbBackColor Property	151
1.3.18.8	TicColor Property	151
1.3.18.9	TicFont Property	151
1.3.18.10	TicOffset Property	151
1.3.18.11	TicTextColor Property	152
1.3.18.12	TicTextOffset Property	152
1.3.18.13	TicWidth Property	152
1.3.18.14	Ticks Property	153
1.3.18.15	TicksColor Property	153
1.3.18.16	TickStyle Property	153
1.3.18.17	TimeAxis Property	153
1.3.18.18	TimeAxisFormat Property	153
1.3.18.19	TimeAxisX Property	154
1.3.18.20	TimeColumnAlignment Property	154
1.3.18.21	TimeFormat Property	154
1.3.18.22	TimeJump Property	154
1.3.18.23	TimeJumpColor Property	155
1.3.18.24	TimeOverlap Property	155
1.3.18.25	TimeOverlapColor Property	155
1.3.18.26	TimeRange Property	155
1.3.18.27	TimeRangeBase Property	155
1.3.18.28	TimeRangeFactor Property	156
1.3.18.29	TimeStamp Property	156
1.3.18.30	TimeZone Property	157
1.3.18.31	TitleCut Property	157
1.3.18.32	Titleline Property	158
1.3.18.33	Toggle Property	158
1.3.18.34	ToleranceHigh Property	158
1.3.18.35	ToleranceLow Property	158
1.3.18.36	ToolbarAlignment Property	158
1.3.18.37	ToolbarButtons Property	158
1.3.18.38	Toolbar Property	159
1.3.18.39	ToolbarHotKeys Property	159
1.3.18.40	ToolTipText Property	160
1.3.18.41	Top Property	160
1.3.18.42	TopConnectedConnectionPointIndex Property	161
1.3.18.43	TopConnectedObjectName Property	161

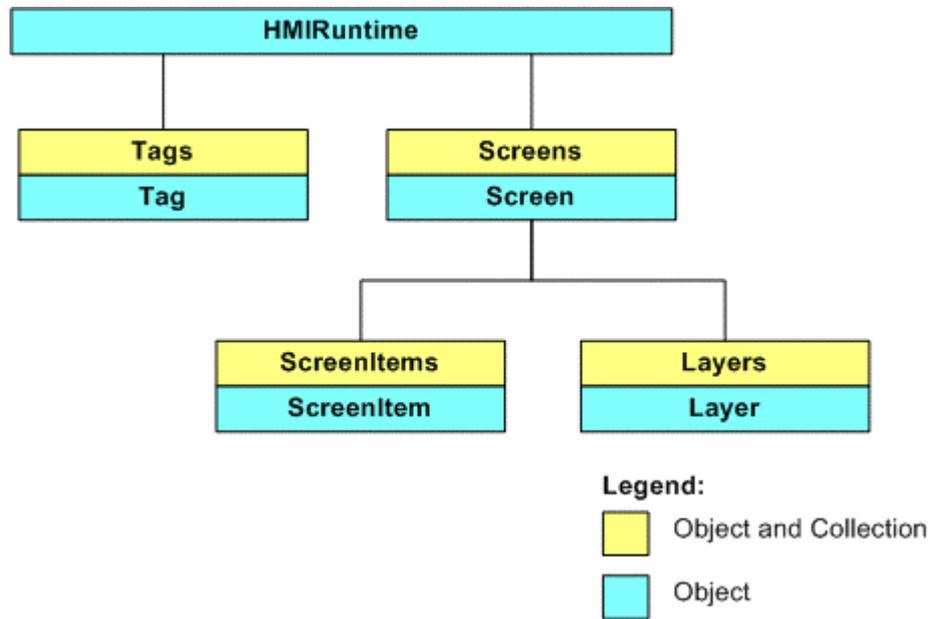
1.3.18.44	Transparent Property.....	161
1.3.18.45	Trend Property.....	161
1.3.18.46	TrendColor Property.....	162
1.3.18.47	Type Property.....	162
1.3.18.48	TypeAlarmHigh Property.....	163
1.3.18.49	TypeAlarmLow Property.....	163
1.3.18.50	TypeLimitHigh4 Property.....	163
1.3.18.51	TypeLimitHigh5 Property.....	163
1.3.18.52	TypeLimitLow4 Property.....	163
1.3.18.53	TypeLimitLow5 Property.....	164
1.3.18.54	TypeToleranceHigh Property.....	164
1.3.18.55	TypeToleranceLow Property.....	164
1.3.18.56	TypeWarningHigh Property.....	164
1.3.18.57	TypeWarningLow Property.....	165
1.3.19	U.....	165
1.3.19.1	UnitColor Property.....	165
1.3.19.2	UnitFont Property.....	165
1.3.19.3	UnitOffset Property.....	165
1.3.19.4	UnitText Property.....	166
1.3.19.5	UnselBGColor Property.....	166
1.3.19.6	UnselTextColor Property.....	166
1.3.19.7	UserValue1 Property.....	166
1.3.19.8	UserValue2 Property.....	166
1.3.19.9	UserValue3 Property.....	166
1.3.19.10	UserValue4 Property.....	166
1.3.19.11	UpdateCycle Property.....	167
1.3.19.12	UpperLimit Property.....	167
1.3.19.13	UpperLimitColor Property.....	167
1.3.19.14	UpperLimitValue Property.....	167
1.3.20	V-Z.....	167
1.3.20.1	ValueColumnAlignment Property.....	167
1.3.20.2	Value Property.....	168
1.3.20.3	ValueMax Property.....	168
1.3.20.4	ValueMin Property.....	168
1.3.20.5	Variable Property.....	168
1.3.20.6	Visible Property.....	168
1.3.20.7	Warning Property.....	169
1.3.20.8	WarningColor Property.....	169
1.3.20.9	WarningHigh Property.....	169
1.3.20.10	WarningLow Property.....	169
1.3.20.11	Width Property.....	170
1.3.20.12	WindowBorder Property.....	170
1.3.20.13	WindowsStyle Property.....	170
1.3.20.14	WindowType Property.....	171
1.3.20.15	WithAxes Property.....	171
1.3.20.16	WithLabels Property.....	171
1.3.20.17	ZeroPoint Property.....	171
1.3.20.18	ZeroPointValue Property.....	171
1.3.20.19	Zoom Property.....	171
1.3.20.20	Zoomable Property.....	172
1.4	Methods.....	172
1.4.1	Activate Method.....	172
1.4.2	Item Method.....	173
1.4.3	Read Method.....	174
1.4.4	Refresh Method.....	177

1.4.5	Stop Method	177
1.4.6	Trace Method	178
1.4.7	Write Method	178

1 VBS Reference

VBS Object Model in WinCC

Referenztablelle



The WinCC object model of the graphic runtime system enables access to the graphic objects and tags in runtime:

Objects

Objects and lists are provided for access to all the objects in the graphic runtime systems: Graphic objects, pictures, layers and tags.

The properties of the individual objects can be used to modify specific graphic objects and tags in runtime , e.g. activating an operating element per mouse click or triggering a color change by modifying a tag value.

Methods

Methods, which are applied to individual objects, can be used to read out tag values for further processing or displaying diagnostics messages in runtime.

1.1 Objects and Lists

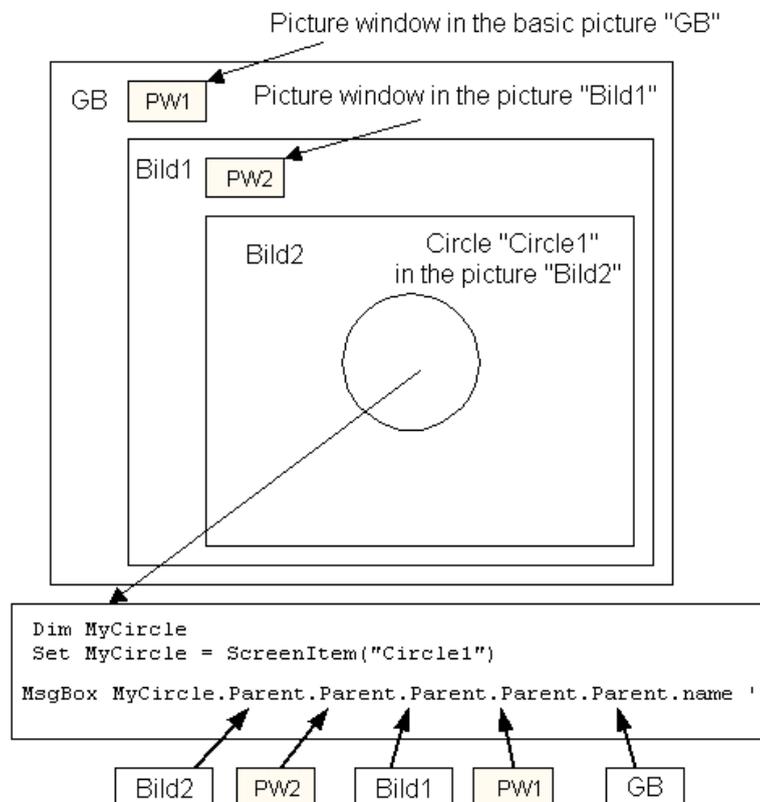
Overview

The objects and lists provided in WinCC object models enables access to graphic objects and tags in runtime.

Navigation in Object Models

Access is made to objects in the VBS object model in hierarchical sequence. If, for example, a picture element is accessed within a picture, access is made to the picture element in the picture via its parent object (the surrounding picture).

Example



Only the basic picture name is issued in this example.

Access to Graphic Objects

In WinCC, access is made to pictures, layers and graphic objects in runtime via the superordinate "HMIRuntime" object. Access to objects and layers is always made via the picture (screen) in which they are contained.

Access to Tags

In WinCC, tags are accessed directly in runtime via the superordinate "HMIRuntime" object. Tag values can be read out or set anew.

Lists

Lists of WinCC object models behave in the same way as standard collections of VBS. Exception: The "Tags" list has no Enum function.

1.1.1 ActiveScreen Object

The "ActiveScreen" object provides a reference for the picture containing the object which currently has the focus.

Usage

The "ActiveScreen" object is used in runtime to address the properties of the picture which contains the currently focussed object.

Example

The following example assigns the "strScrName" tag to the name of the current picture:

```
'VBS118
Dim strScrName
strScrName = HMIRuntime.ActiveScreen.Objectname
MsgBox strScrName
```

1.1.2 ActiveScreenItem Object

The "ActiveScreenItem" object provides a reference to the object which currently has the focus.

Usage

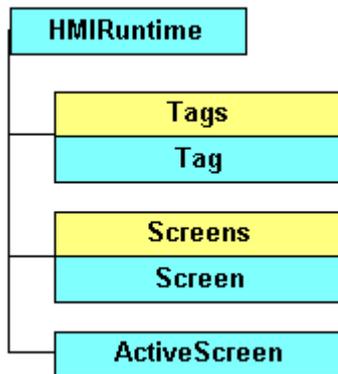
The "ActiveScreenItem" object is used in runtime in order to address the properties of the object which currently has the focus.

Example

The following example displays the name of the object in the "ScreenWindow1" picture which has the focus:

```
'VBS119
Dim objScreen
Set objScreen = HMIRuntime.Screens("ScreenWindow1")
MsgBox objScreen.ActiveScreenItem.Objectname
```

1.1.3 HMIRuntime Object



The HMIRuntime object represents the graphic runtime environment.

Usage

The "HMIRuntime" object can be used for the following, for example:

- Read or define the current runtime language ("Language" property).
- Read or define the name of the current basic picture ("ActiveScreen" property).
- Access tags ("Tags" property).
- Exit runtime ("Stop" method).
- Display messages in a diagnostics window ("Trace" method).

Example

The following command terminates WinCC runtime:

```
'VBS3  
HMIRuntime.Stop
```

1.1.4 Item Object

The "Item" object provides a reference to the current object.

Usage

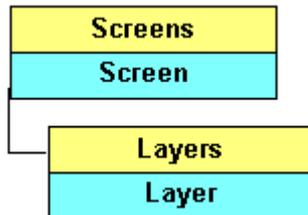
The "Item" object is used, for example, to address the properties of the object currently selected in Graphics Designer.

Example

In the following example, a rectangle has been created. When the object has been selected, all the properties of the current object can be set a background color red:

```
Item.BackColor = (255,0,0)
```

1.1.5 Layer Object



The Layer returns the result of access to the Layers list.

Parent Object

Picture, in which the picture layer is.

Usage

Depending on certain events, the Layer object can be used to obtain access to the properties of a complete layer in order, for example, to hide or unhide a layer with operating elements according to the operator authorization.

The "Layer" object can be used to:

- To activate or deactivate the visualization of a layer ("Visible" property).
- To read out the name of a layer ("Name" property).

Note:

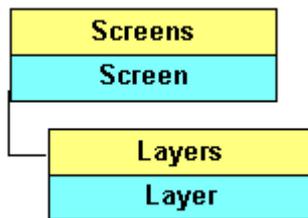
The Layer Property defines the layer in which the object is located. Layer 0 is displayed as layer "0". During access using VBS, the layers are counted up from 1. Therefore, layer 1 must be referred to as layers(2).

Example

In the following example, Layer 1 is set invisible:

```
'VBS4
Layers(2).Visible = vbFalse
```

1.1.6 Layers Object (List)



The Layers listing enables access to all 32 layers of the graphical runtime system.

Parent Object

Picture, in which the picture layer is.

Usage

The "Layers" list can be used to:

- Process all the layers in the listing ("_NewEnum" property).
- Count all the layers contained in the list ("Count" property).
- Process a layer from the list ("Item" method).

The properties are standard properties and methods of a collection and are not described in detail in the WinCC documentation.

1.1.7 Parent Object

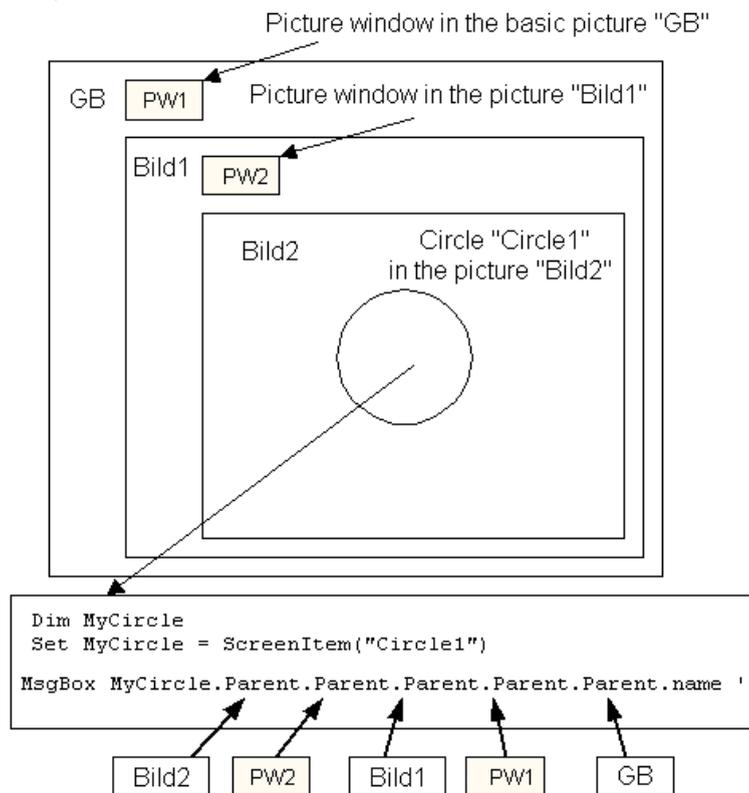
Access is made to objects in the VBS object model in hierarchical sequence. If, for example, a picture element is accessed within a picture, access is made to the picture element in the picture via its parent object (the surrounding picture).

The Parent object provides a reference to the object that is one layer higher in the object hierarchy than that originally referenced.

Usage

The Parent object can be used as often as required within an object hierarchy. The following section provides a systematic of how to access all the elements in a

hierarchy:



The Command

`MsgBox MyCircle.Parent`

that returns the name of "Picture2" which lies one layer higher in the object hierarchy than the original ScreenItem object "Circle1".

If "Parent" is used 3 times, for example, you access three layers higher in the object hierarchy:

`MsgBox MyCircle.Parent.Parent.Parent`
this displays the name of Picture 1, because

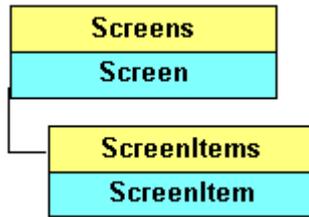
- the original reference the ScreenItem "Circle1"
- "Circle1" is in "Picture2" (Layer 1)
- "Picture2" is in Picture Window2 "BF2" (Layer 1)
- 2nd "BF2" is in "Picture 1" (Layer 1)

Example

In the following example, the object name of the Parent object is displayed:

```
'VBS120
Dim objCircle
Set objCircle = HMIRuntime.Screens("ScreenWindow1").ScreenItems("Circle1")
MsgBox objCircle.Parent.ObjectName
```

1.1.8 ScreenItem Object



The ScreenItem object returns the result of access to the ScreenItems list.

Parent Object

Picture containing the picture element.

Usage

The ScreenItem object can be used to access the properties of graphic objects within a picture according to certain events.

The "ScreenItem" object can be used for the following, for example:

- To activate or deactivate the visualization of an object ("Visible" property).
- To release or block the operation of an object ("Enabled" property).
- Change the width and height of an object ("Height" and "Width" properties).
- Change the position of an object ("Top" and "Left" properties).
- Read out and define a layer in which a graphic object is located ("Layer" property).
- Read out or define the name of a graphic object ("ObjectName" property).
- Define a reference to the superordinate picture ("Parent" property).

Possible features of ScreenItem

The "ScreenItem" object can contain the following object types:

Standard objects	Smart Objects	Windows Objects	Controls	Continue
Ellipse	3D-Bar	Button	WinCC Alarm Control	Customized object
Ellipse Arc	Application Window	Check Box	WinCC Digital/Analog Clock	Group object
Ellipse Segment	Bar	Option Group	WinCC DXF Control	
Circle	Picture Window	Round Button	WinCC Function Trend Control	
Circular Arc	Control	Slider	WinCC Gauge Control	
Pie Segment	I/O Field		WinCC Online Table Control	

Standard objects	Smart Objects	Windows Objects	Controls	Continue
Line	Graphic Object		WinCC Online Trend Control	
Polygon	OLE Object		WinCC Push Button Control	
Polyline	Group Display		WinCC Slider Control	
Rectangle	Text List		HMI Symbol Library 1.3	
Rounded Rectangle	Status Display			
Static Text				
Connector				

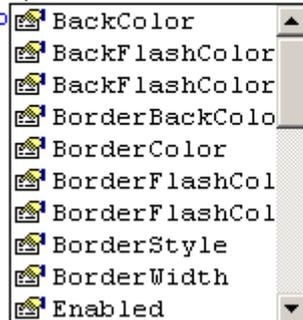
Detailed s of the individual object types is provided under "ScreenItem Object Types". The ScreenItem object's "Type" property can be used to address the object types via the VBS Type ID.

Object properties

The "ScreenItem" object has different properties according to the features. The following section describes the properties which all ScreenItem object types have:

When a specific object type is addressed, certain further properties are added to the standard properties:

```
Sub OnClick(ByVal Item)
Dim obj
Set obj = ScreenItems("Rechteck1").Height
Set obj.|
End Sub
```



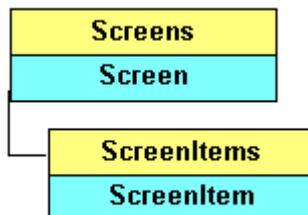
The additional properties are indicated in the s of the individual object types.

Example

In the following example, the radius of a circle is defined in runtime per mouse click on 2:

```
Sub OnClick (ByVal Item)
'VBS5
Dim objCircle
Set objCircle= ScreenItems("Circle1")
objCircle.Radius = 2
End Sub
```

1.1.9 ScreenItems Object (List)



The "ScreenItems" list can be used to reference an object in the picture.

Parent Object

Picture containing the picture element.

Usage

The "ScreenItems" list can be used to:

- To display or edit all objects in the list (i.e. all objects within a picture) ("_NewEnum" property).
- To count the objects in a picture ("Count" property).
- To process a specific object in the list ("Item" method).

The properties are standard properties and methods of a collection and are not described in detail in the WinCC documentation.

Special features of the ScreenItem object

If an external control (ActiveX control or OLE object) is embedded in WinCC, it is possible that the properties of the embedded controls have the same name with the general properties of the ScreenItem object. In such cases, the ScreenItem properties have priority.

The properties of the embedded controls can also be addressed via the "object" property:

The "object" property is only provided by ActiveX controls and OLE objects.

Example:

```
'Control1 ist ein eingebettetes ActiveX-Control mit der Eigenschaft type
Dim Control
Set Control=ScreenItems("Control1")
Control.object.type
```

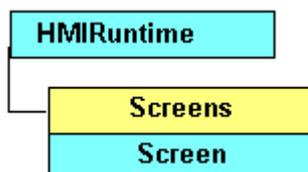
```
'Control1 ist ein WinCC-Control
Dim Control
Set Control=ScreenItems("Control1")
Control.type
```

Example

In the following example, the name of the objects in the current picture are displayed in a message box:

```
Sub OnClick(ByVal Item)
'VBS6
Dim lngAnswer
Dim lngIndex
lngIndex = 1
For lngIndex = 1 To ScreenItems.Count
lngAnswer = MsgBox(ScreenItems(lngIndex).Objectname, vbOKCancel)
If vbCancel = lngAnswer Then Exit For
Next
End Sub
```

1.1.10 Screen Object



The Screen object returns the result of access to the Screen list. All the properties and methods of this object can also be edited directly in runtime. The "Screen" object represents a WinCC picture in runtime and contains all the properties of the picture document and picture view.

The "Screen" object also contains the following:

- A list of all the graphic objects contained in the addressed picture which can be addressed by the "ScreenItems" object.
- A list of all the layers contained in the addressed picture which can be addressed by the "Layers" object.

Parent Object

A picture window in which the Screen object is embedded.

When the Screen object is the basic picture, the Parent object is not defined and set to zero.

Usage

The "Screen" object can be used for the following, for example:

- To release or block the operation of a screen ("Enabled" property).
- Change the width and height of a screen ("Height" and "Width" properties).
- Zoom a picture ("Zoom" property).
- Modify the fill pattern, background color and fill pattern color ("Fillstyle", "BackColor" and "FillColor" properties).

Note:

If a Change Picture is executed, all the open references are invalid for pictures no longer open. It is then no longer possible to work with these references.

Example

In the following example, the width of the first picture in runtime is increased by 20 pixels:

```
'VBS7
Dim objScreen
Set objScreen = HMIRuntime.Screens(1)
MsgBox "Screen width before changing: " & objScreen.Width
objScreen.Width = objScreen.Width + 20
MsgBox "Screen width after changing: " & objScreen.Width
```

Notes on Cross References

All the pictures which are addressed with the standard formulation

```
HMIRuntime.BaseScreenName = "Screenname"
```

are automatically compiled by the CrossReference of WinCC and then listed in the picture properties.

If pictures are addressed with different formulations in the code, this can be notified by the following section of the CrossReference:

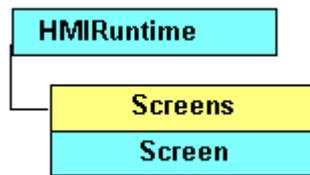
```
' WINCC:SCREENNAME_SECTION_START
Const ScreenNameInAction = "ScreenName"
' WINCC:SCREENNAME_SECTION_END
```

The section can be inserted in VBS actions as often as required.

Note:

Always enter picture names without the extension "PDL" for reasons of compatibility with future versions.

1.1.11 Screens Object (List)



By using the picture window technique, several windows can be opened simultaneously in WinCC runtime but only one basic picture exists. The "Screens" list enables access to all open pictures in runtime using the picture names. The Screens list contains all invisible pictures.

Usage

When configuring a multi-user project, it is essential to specify the server prefix to access a picture which is not on the local computer.

The "Screens" list can be used to:

- Display or edit all the pictures within the list ("_NewEnum" property).
- To count the pictures in a project ("Count" property).
- To process a specific picture in the list ("Item" method).
- Initiate new drawing of all visible pictures ("Refresh" method).

The properties are standard properties and methods of a collection and are not described in detail in the WinCC documentation.

The access code, required in the VBS environment in the HMIRuntime.Screens(<access code>) instruction, must fulfill the syntax requirements:

```
[<Grundbildname>].<Bildfenstername>[:<Bildname>] ...
.<Bildfenstername>[:<Bildname>]
```

This means:

- The access code expresses the picture hierarchy.
- The picture names in the code can be omitted at any point.
- The "AccessPath" property of the "Screen" object corresponds to the full access code.
- Always enter picture names without the extension "PDL" for reasons of compatibility with future versions.
- The basic picture can be addressed by the access code "".

In addition, it has been defined that the basic picture can be addressed with Index1.

Examples

The pictures are addressed by the hierarchy information in the list. There are two options here, with or without use of the picture name. In the following examples, a basic picture "BaseScreenName" is configured with a picture window "ScreenWindow". The picture window contains the picture "ScreenName".

Addressing with the picture name

```
'VBS8  
Set objScreen =  
HMIRuntime.Screens("BaseScreenName.ScreenWindow:ScreenName")
```

Addressing without the picture name

```
'VBS9  
Set objScreen = HMIRuntime.Screens("ScreenWindow")
```

Referencing the basic picture in various ways

```
'VBS10  
Set objScreen = HMIRuntime.Screens(1)  
'VBS11  
Set objScreen = HMIRuntime.Screens("")  
'VBS12  
Set objScreen = HMIRuntime.Screens("BaseScreenName")
```

1.1.12 Tag Object



A tag object is returned via the "Tags" list. A tag object can be used to address all the properties and methods of a tag.

When creating a tag object, all the properties are installed with the following values:

- Value = VT_EMPTY
- Name = Tag name
- QualityCode = BAD NON-SPECIFIC
- TimeStamp = 0
- LastError = 0
- Error = " "

Note:

A summary of the possible Quality Codes is provided in the WinCC Information System under the heading "Communication" > "Diagnostics" or "Communication" > "Quality Codes".

Usage

The "Tag" object can be used to:

- Read out information on the tag ("Name", "QualityCode", "TimeStamp", "LastError" and "Error" properties)
- Set a value for a tag ("Write" method, "Value" property)
- Read a value for a tag ("Read" method, "Value" property)

Read the value of a "Tag1" tag:

```
'VBS13
Dim objTag
Set objTag = HMIRuntime.Tags("Tag1")
objTag.Read()
MsgBox objTag.Value
```

Declaration of tags in WinCC

Always define internal tags in VBScript using the "Dim" instruction in order to prevent writing tags wrongly.

When creating a new action, the "Option explicit" instruction is automatically entered in the declaration area and cannot be deleted.

Do not use the "Option explicit" instruction in the code because it may cause runtime errors.

Example: Declaration of a VBScript "IngVar" tag:

```
'VBS14
Dim IngVar
IngVar = 5
MsgBox IngVar
```

Note:

Tag names must not contain any special characters.

Note:

Please note that when creating a tag, it must not contain a value (Value = VT_EMPTY). Initialize the tags after declaration with the corresponding value.

Notes on Cross References

All the pictures which are addressed with the standard formulation

HMIRuntime.Tags("Tagname")
are automatically compiled by the CrossReference of WinCC and then listed in the picture properties.

If tags are addressed with different formulations in the code, this can be notified by the following section of the CrossReference:

```
' WINCC:TAGNAME_SECTION_START  
Const TagNameInAction = "TagName"  
' WINCC:TAGNAME_SECTION_END
```

The section can be inserted in VBS actions as often as required.

Note:

It is not possible to guarantee the compilation of combined tag names from the CrossReference.

1.1.13 Tags Object (List)



The "Tags" list enables access to tags in WinCC runtime. The result of access to the "Tags" list is returned by an object of the type "Tag". The Tag object can be used to access all the tag properties and methods.

Note:

"Tags" is a list with a restricted functional scope. The tags in the list cannot be accessed via the index but only by using the tag names. The standard methods `get_Count` and `get_NewEnum` cannot be used in the Tags list.

Usage

Tags in the list are accessed via:

```
HMIRuntime.Tags("Tagname")
```

The Tags list is used to declare tags (tag objects) for read and write access. In order that the read and write access can be executed without any errors, the corresponding tags must be available in the WinCC Tag Management.

Using VBScript in WinCC it is possible to address tags directly by their name to define and read values. If additional tag properties are required, such as requesting the quality code or a time stamp or executing error measures, the tag must always be addressed via the Tags list. The tag object returned enables access to all tag properties and methods.

Example

There are two options when creating tags:

- With specification of the server prefix: For tags in multi-user systems which are not stored locally.
- Direct use of the tag name: For tags stored locally on the computer.

Specification of the server prefix

```
'VBS15  
Dim objTag  
Set objTag = HMIRuntime.Tags("Serverprefix::Tagname")
```

If the server prefix is entered directly, the "ServerPrefix" property is assigned the corresponding value.

Specification of the tag name

```
'VBS16  
Dim objTag  
Set objTag = HMIRuntime.Tags("Tagname")
```

If just the tag name is used, the "ServerPrefix" and "TagPrefix" properties are assigned the values from the current context (current picture window).

1.2 Object Type of the ScreenItem Object

Introduction

The following section lists all the available types of the "ScreenItem" object.

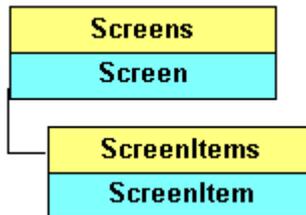
The features of the "ScreenItem" object represent all the graphic objects available in WinCC Graphics Designer.

The object types are divided into the following groups according to their arrangement in Graphics Designer:

There are also the object types

1.2.1 Standard objects

1.2.1.1 Ellipse



Object Type of the ScreenItem Object Represents the graphic object "Ellipse"

Type Identifier in VBS

HMIIEllipse

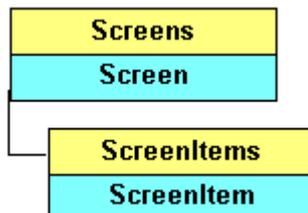
Usage

In the following example, the object with the name "Ellipse1" is moved 10 pixels to the right:

```
'VBS17
Dim objEllipse
Set objEllipse = ScreenItems("Ellipse1")
objEllipse.Left = objEllipse.Left + 10
```

This object type has the following properties:

1.2.1.2 Ellipse Arc



Object Type of the ScreenItem Object Represents the graphic object "Ellipse Arc"

Type Identifier in VBS

HMIIEllipticalArc

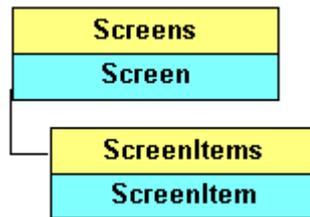
Usage

In the following example, the object with the name "EllipseArc1" is moved 10 pixels to the right:

```
'VBS18
Dim objEllipseArc
Set objEllipseArc = ScreenItems("EllipseArc1")
objEllipseArc.Left = objEllipseArc.Left + 10
```

This object type has the following properties:

1.2.1.3 Ellipse Segment



Object Type of the ScreenItem Object Represents the graphic object "Ellipse Segment"

Type Identifier in VBS

HMIEllipseSegment

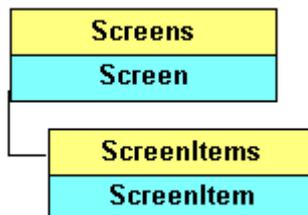
Usage

In the following example, the object with the name "EllipseSegment1" is moved 10 pixels to the right:

```
'VBS19
Dim objEllipseSeg
Set objEllipseSeg = ScreenItems("EllipseSegment1")
objEllipseSeg.Left = objEllipseSeg.Left + 10
```

This object type has the following properties:

1.2.1.4 Circle



Object Type of the ScreenItem Object Represents the graphic object "Circle"

Type Identifier in VBS

HMICircle

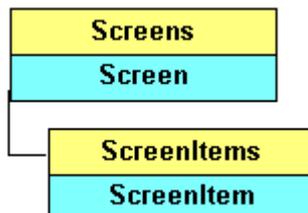
Usage

In the following example, the object with the name "Circle1" is moved 10 pixels to the right:

```
'VBS20
Dim objCircle
Set objCircle = ScreenItems("Circle1")
objCircle.Left = objCircle.Left + 10
```

This object type has the following properties:

1.2.1.5 Circular Arc



Object Type of the ScreenItem Object Represents the graphic object "Circular Arc"

Type Identifier in VBS

HMICircularArc

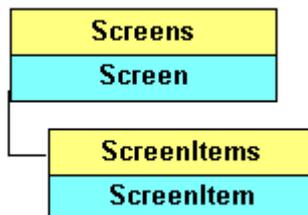
Usage

In the following example, the object with the name "CircularArc1" is moved 10 pixels to the right:

```
'VBS21
Dim objCircularArc
Set objCircularArc = ScreenItems("CircularArc1")
objCircularArc.Left = objCircularArc.Left + 10
```

This object type has the following properties:

1.2.1.6 Pie Segment



Object Type of the ScreenItem Object Represents the graphic object "Pie Segment"

Type Identifier in VBS

HMICircleSegment

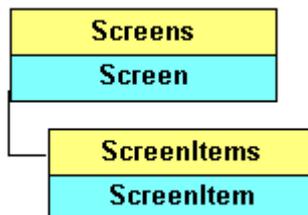
Usage

In the following example, the object with the name "PieSegment1" is moved 10 pixels to the right:

```
'VBS22
Dim objCircleSeg
Set objCircleSeg = ScreenItems("PieSegment1")
objCircleSeg.Left = objCircleSeg.Left + 10
```

This object type has the following properties:

1.2.1.7 Line



Object Type of the ScreenItem Object Represents the graphic object "Line"

Type Identifier in VBS

HMLLine

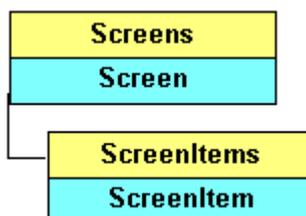
Usage

In the following example, the object with the name "Line1" is moved 10 pixels to the right:

```
'VBS23
Dim objLine
Set objLine = ScreenItems("Line1")
objLine.Left = objLine.Left + 10
```

This object type has the following properties:

1.2.1.8 Polygon



Object Type of the ScreenItem Object Represents the graphic object "Polygon"

Type Identifier in VBS

HMIPolygon

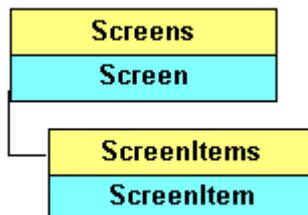
Usage

In the following example, the object with the name "Polygon1" is moved 10 pixels to the right:

```
'VBS24
Dim objPolygon
Set objPolygon = ScreenItems("Polygon1")
objPolygon.Left = objPolygon.Left + 10
```

This object type has the following properties:

1.2.1.9 Polyline



Object Type of the ScreenItem Object Represents the graphic object "Polyline"

Type Identifier in VBS

HMIPolyLine

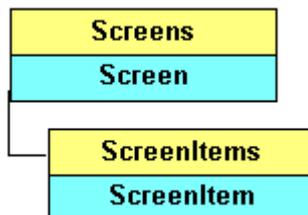
Usage

In the following example, the object with the name "Polyline1" is moved 10 pixels to the right:

```
'VBS25
Dim objPolyline
Set objPolyline = ScreenItems("Polyline1")
objPolyline.Left = objPolyline.Left + 10
```

This object type has the following properties:

1.2.1.10 Rectangle



Object Type of the ScreenItem Object Represents the graphic object "Rectangle"

Type Identifier in VBS

HMIRectangle

Usage

In the following example, the object with the name "Rectangle1" is moved 10 pixels to the right:

```
'VBS26
Dim objRectangle
Set objRectangle = ScreenItems("Rectangle1")
objRectangle.Left = objRectangle.Left + 10
```

Notes on Error Handling

The rectangle and rounded rectangle are mapped to an "HMIRectangle" type in the object model. Since the two objects have different properties, the availability of the property (dynamic type compilation in runtime) should be queried via an exception measure. The exception measure is activated for the corresponding procedure by the following instruction:

```
On Error Resume Next
```

The instruction causes the VBScript engine to initiate the follow-on command in the case of a runtime error.

The error code can subsequently be checked using the Err object. In order to deactivate the handling of runtime errors in scripts, use the following command:

```
On Error Goto 0
```

Handling errors always relates to the procedure layer. If a script in a procedure causes an error, VBScript checks whether an error handling measure is implemented in this layer. If not, control is transferred one layer up (to the calling procedure). If there is no error handling measure here either, the control is transferred yet another layer up. This continues until either the top module level is

reached or the code for runtime error handling is located. If the activation of the runtime error handling fails, the control is transferred to the top level on the internal VBScript runtime error handling. This opens an error dialog and stops the script.

The "On Error Resume Next" command can be installed on all layers (i.e. also in procedures). When the error handling measure is use, it can basically be determined whether the user is actually using the required implementation type.

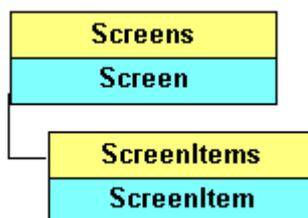
In addition, it can be ensured that there is no termination of execution due to a faulty access to the object.

Examples of error handling

```
Sub OnClick(ByVal Item)
'VBS27
Dim objScreenItem
'
'Activation of errorhandling:
On Error Resume Next
For Each objScreenItem In ScreenItems
If "HMIRectangle" = objScreenItem.Type Then
'
'=== Property "RoundCornerHeight" only available for RoundedRectangle
objScreenItem.RoundCornerHeight = objScreenItem.RoundCornerHeight * 2
If 0 <> Err.Number Then
HMIRuntime.Trace objScreenItem.Name & ": no RoundedRectangle" & vbCrLf
'
'Delete error message
Err.Clear
End If
End If
Next
On Error Goto 0 'Deactivation of errorhandling
End Sub
```

This object type has the following properties:

1.2.1.11 Rounded Rectangle



Object Type of the ScreenItem Object Represents the graphic object "Rounded Rectangle"

Type Identifier in VBS

HMIRectangle

Usage

In the following example, the object with the name "RoundedRectangle1" is moved 10 pixels to the right:

```
'VBS28
Dim objRoundedRectangle
Set objRoundedRectangle = ScreenItems("RoundedRectangle1")
objRoundedRectangle.Left = objRoundedRectangle.Left + 10
```

Notes on Error Handling

The rectangle and rounded rectangle are mapped to an "HMIRectangle" type in the object model. Since the two objects have different properties, the availability of the property (dynamic type compilation in runtime) should be queried via an exception measure. The exception measure is activated for the corresponding procedure by the following instruction:

On Error Resume Next

The instruction causes the VBScript engine to initiate the follow-on command in the case of a runtime error.

The error code can subsequently be checked using the Err object. In order to deactivate the handling of runtime errors in scripts, use the following command:

On Error Goto 0

Handling errors always relates to the procedure layer. If a script in a procedure causes an error, VBScript checks whether an error handling measure is implemented in this layer. If not, control is transferred one layer up (to the calling procedure). If there is no error handling measure here either, the control is transferred yet another layer up. This continues until either the top module level is reached or the code for runtime error handling is located. If the activation of the runtime error handling fails, the control is transferred to the top level on the internal VBScript runtime error handling. This opens an error dialog and stops the script.

The "On Error Resume Next" command can be installed on all layers (i.e. also in procedures). When the error handling measure is use, it can basically be determined whether the user is actually using the required implementation type.

In addition, it can be ensured that there is no termination of execution due to a faulty access to the object.

Examples of error handling

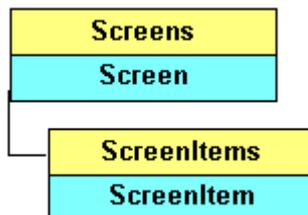
```

Sub OnClick(ByVal Item)
'VBS29
Dim objScreenItem
On Error Resume Next 'Activation of errorhandling
For Each objScreenItem In ScreenItems
If "HMIRectangle" = objScreenItem.Type Then
'
'=== Property "RoundCornerHeight" available only for RoundedRectangle
objScreenItem.RoundCornerHeight = objScreenItem.RoundCornerHeight * 2
If 0 <> Err.Number Then
HMIRuntime.Trace objScreenItem.ObjectName & ": no RoundedRectangle" &
vbCrLf
Err.Clear 'Delete errormessage
End If
End If
Next
On Error Goto 0 'Deactivation of errorhandling
End Sub

```

This object type has the following properties:

1.2.1.12 Static Text



Object Type of the ScreenItem Object Represents the graphic object "Static Text"

Type Identifier in VBS

HMITextField

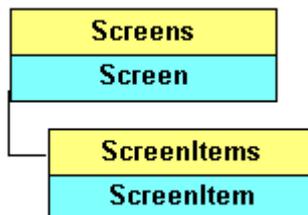
Usage

In the following example, the object with the name "StaticText1" is moved 10 pixels to the right:

```
'VBS30
Dim objStaticText
Set objStaticText = ScreenItems("StaticText1")
objStaticText.Left = objStaticText.Left + 10
```

This object type has the following properties:

1.2.1.13 Connector



Object Type of the ScreenItem Object Represents the graphic object "Connector"

Type Identifier in VBS

HMIConnector

Usage

In the following example, the object with the name "Connector1" is moved 10 pixels to the right:

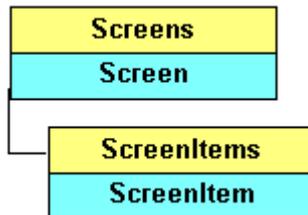
```
'VBS31
Dim objConnector
Set objConnector = ScreenItems("Connector1")
objConnector.Left = objConnector.Left + 10
```

This object type has the following properties:

1.2.2 Smart Objects

Smart objects available

1.2.2.1 3D-Bar



Object Type of the ScreenItem Object Represents the graphic object "3D-Bar"

Type Identifier in VBS

HMIBar

Usage

In the following example, the object with the name "3DBar1" is moved 310 pixels to the right:

```
'VBS32
Dim objBar
Set objBar = ScreenItems("3DBar1")
objBar.Left = objBar.Left + 10
```

Notes on Error Handling

Bars and 3D bars are imaged in the object model on a "HMIBar" type. Since the two objects have different properties, the availability of the property (dynamic type compilation in runtime) should be queried via an exception measure. The exception measure is activated for the corresponding procedure by the following instruction:

```
On Error Resume Next
```

The instruction causes the VBScript engine to initiate the follow-on command in the case of a runtime error.

The error code can subsequently be checked using the Err object. In order to deactivate the handling of runtime errors in scripts, use the following command:

```
On Error Goto 0
```

Handling errors always relates to the procedure layer. If a script in a procedure causes an error, VBScript checks whether an error handling measure is implemented in this layer. If not, control is transferred one layer up (to the calling procedure). If there is no error handling measure here either, the control is transferred yet another layer up. This continues until either the top module level is reached or the code for runtime error handling is located. If the activation of the runtime error handling fails, the control is transferred to the top level on the internal VBScript runtime error handling. This opens an error dialog and stops the script.

The "On Error Resume Next" command can be installed on all layers (i.e. also in procedures). When the error handling measure is use, it can basically be determined whether the user is actually using the required implementation type.

In addition, it can be ensured that there is no termination of execution due to a faulty access to the object.

Examples of error handling

```
'VBS148
```

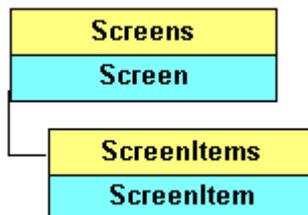
```

Sub OnClick(ByVal Item)
Dim objScreenItem
'
'Activation of errorhandling:
On Error Resume Next
For Each objScreenItem In ScreenItems
If "HMIBar" = objScreenItem.Type Then
'
'=== Property "Layer00Value" only available for 3D bar
objScreenItem.Layer00Value = objScreenItem.Layer00Value * 2
If 0 <> Err.Number Then
HMIRuntime.Trace objScreenItem.Name & ": no 3D bar" & vbCrLf
'
'Delete error message
Err.Clear
End If
End If
Next
On Error Goto 0 'Deactivation of errorhandling
End Sub

```

This object type has the following properties:

1.2.2.2 Application Window



Object Type of the ScreenItem Object Represents the graphic object "Application Window"

Type Identifier in VBS

HMIApplicationWindow

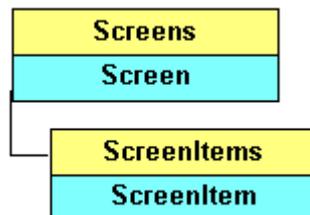
Usage

In the following example, the object with the name "ApplicationWindow1" is moved 10 pixels to the right:

```
'VBS33
Dim objAppWindow
Set objAppWindow = ScreenItems("ApplicationWindow1")
objAppWindow.Left = objAppWindow.Left + 10
```

This object type has the following properties:

1.2.2.3 Bar



Object Type of the ScreenItem Object Represents the graphic object "Bar"

Type Identifier in VBS

HMIBar

Usage

In the following example, the object with the name "Bar1" is moved 10 pixels to the right:

```
'VBS34
Dim objBar
Set objBar = ScreenItems("Bar1")
objBar.Left = objBar.Left + 10
```

Notes on Error Handling

Bars and 3D bars are imaged in the object model on a "HMIBar" type. Since the two objects have different properties, the availability of the property (dynamic type compilation in runtime) should be queried via an exception measure. The exception measure is activated for the corresponding procedure by the following instruction:

On Error Resume Next

The instruction causes the VBScript engine to initiate the follow-on command in the case of a runtime error.

The error code can subsequently be checked using the Err object. In order to deactivate the handling of runtime errors in scripts, use the following command:

On Error Goto 0

Handling errors always relates to the procedure layer. If a script in a procedure causes an error, VBScript checks whether an error handling measure is implemented in this layer. If not, control is transferred one layer up (to the calling procedure). If there is no error handling measure here either, the control is transferred yet another layer up. This continues until either the top module level is reached or the code for runtime error handling is located. If the activation of the

runtime error handling fails, the control is transferred to the top level on the internal VBScript runtime error handling. This opens an error dialog and stops the script.

The "On Error Resume Next" command can be installed on all layers (i.e. also in procedures). When the error handling measure is use, it can basically be determined whether the user is actually using the required implementation type.

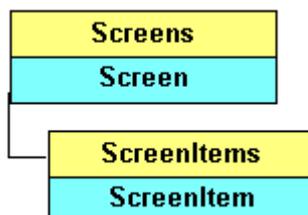
In addition, it can be ensured that there is no termination of execution due to a faulty access to the object.

Examples of error handling

```
'VBS147
Sub OnClick(ByVal Item)
Dim objScreenItem
'
'Activation of errorhandling:
On Error Resume Next
For Each objScreenItem In ScreenItems
If "HMIBar" = objScreenItem.Type Then
'
'=== Property "LimitHigh4" only available for bar
objScreenItem.LimitHigh4 = objScreenItem.LimitHigh4 * 2
If 0 <> Err.Number Then
HMIRuntime.Trace objScreenItem.Name & ": no bar" & vbCrLf
'
'Delete error message
Err.Clear
End If
End If
Next
On Error Goto 0 'Deactivation of errorhandling
End Sub
```

This object type has the following properties:

1.2.2.4 Picture Window



Object Type of the ScreenItem Object Represents the graphic object "Picture Window"

Type Identifier in VBS

HMIScreenWindow

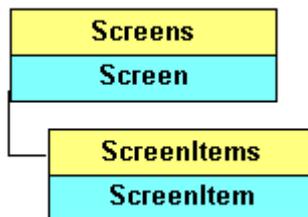
Usage

In the following example, the object with the name "ScreenWindow1" is moved 10 pixels to the right:

```
'VBS35
Dim objScrWindow
Set objScrWindow = ScreenItems("ScreenWindow1")
objScrWindow.Left = objScrWindow.Left + 10
```

This object type has the following properties:

1.2.2.5 Control



Object Type of the ScreenItem Object Represents the graphic object "Control"

The Control object type always assumes the properties of the Control type selected. In the case of controls provided by WinCC, the properties are indicated under the of the corresponding Control.

In the case of controls from external suppliers, the control properties are supplied and thus not a part of this . However, the control properties can be queried using the "Item" property.

Type Identifier in VBS

Special WinCC type s or version-independent ProgID

Usage

In the following example, the object with the name "Control1" is moved 10 pixels to the right:

```
'VBS36
Dim objControl
Set objControl = ScreenItems("Control1")
objControl.Left = objControl.Left + 10
```

Special feature

The controls provided by WinCC return a special ID as the type. It can be found under the topic "Type Identification in VBS in the individual s of the WinCC Controls.

Use of Controls from External Suppliers

In the case of non-WinCC controls, the version-independent ProgID is returned as the type.

It is possible to determine the version-dependent ProgID or "User friendly Name" from the ProgID: In the following example, "Control1" is a control embedded in the picture which already returns the version-independent ProgID as a result of the Type property.

Note:

Since not every Control has a version-dependent ProgID, an error handling measure should be integrated to query the version-dependent ProgID or UserFriendlyName. If no error handling is used, the code is terminated immediately without any result when no ProgID is found.

Determine the version-dependent ProgID as follows:

```
'VBS37
Dim objControl
Dim strCurrentVersion
Set objControl = ScreenItems("Control1")
strCurrentVersion = CreateObject("WScript.Shell").RegRead("HKCR\" &
objControl.Type & "\CurVer")
MsgBox strCurrentVersion
```

Note:

In order that example above works, a multimedia control should be inserted in the picture.

Determine the User Friendly Name as follows:

```
'VBS38
Dim objControl
Dim strFriendlyName
Set objControl = ScreenItems("Control1")
strFriendlyName = CreateObject("WScript.Shell").RegRead("HKCR\" &
objControl.Type & "\")
MsgBox strFriendlyName
```

Note:

In order that example above works, a multimedia control should be inserted in the picture.

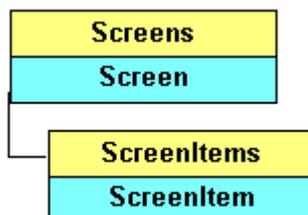
If a non-WinCC control is used, it is possible that the properties provided by the control have the same names as the general ScreenItem properties. In such cases, the ScreenItem properties have priority. The "hidden" properties of an external control supplier can be accessed using the additional "object" property. Address the properties of an external control supplier as follows:

Control.object.type

Only use the form

Control.type

In the case of identical names, the properties of the ScreenItem object are used.

1.2.2.6 I/O Field

Object Type of the ScreenItem Object Represents the graphic object "I/O Field"

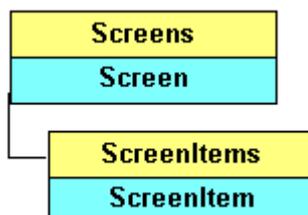
Type Identifier in VBS

HMIIField

Usage

In the following example, the object with the name "IOField1" is moved 10 pixels to the right:

```
'VBS39
Dim objIOField
Set objIOField = ScreenItems("IOField1")
objIOField.Left = objIOField.Left + 10
```

1.2.2.7 Graphic Object

Object Type of the ScreenItem Object Represents the graphic object "Graphic Object"

Type Identifier in VBS

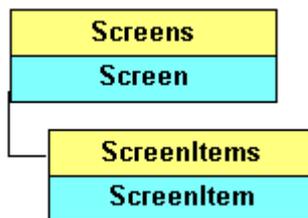
HMIGraphicView

Usage

In the following example, the object with the name "GraphicObject1" is moved 10 pixels to the right:

```
'VBS40
Dim objGraphicView
Set objGraphicView= ScreenItems("GraphicObject1")
objGraphicView.Left = objGraphicView.Left + 10
```

1.2.2.8 OLE Object



Object Type of the ScreenItem Object Represents the graphic object "OLE Object"
The return value is a STRING type.

Type Identifier in VBS

Version-independent ProgID

Usage

In the following example, the object with the name "OLEElement1" is moved 10 pixels to the right:

```
'VBS41
Dim objOLEElement
Set objOLEElement = ScreenItems("OLEElement1")
objOLEElement.Left = objOLEElement.Left + 10
```

Special feature

In the case of OLE objects, the version-independent ProgID is returned as the type.

It is possible to determine the version-dependent ProgID or "User friendly Name" from the ProgID: In the following example, "OLEObject1" is a control embedded in the picture which already returns the version-independent ProgID as a result of the Type property.

Note:

Since not every Control has a version-dependent ProgID, an error handling measure should be integrated to query the version-dependent ProgID or UserFriendlyName. If no error handling is used, the code is terminated immediately without any result when no ProgID is found.

Determine the version-dependent ProgID as follows:

```
'VBS42
Dim objControl
Dim strCurrentVersion
Set objControl = ScreenItems("OLEElement1")
strCurrentVersion = CreateObject("WScript.Shell").RegRead("HKCR\" &
objControl.Type & "\CurVer")
MsgBox strCurrentVersion
```

Note:

In order that the example above works, a Word document should be embedded in the picture as an OLE object.

Determine the User Friendly Name as follows:

```
'VBS43
Dim objControl
Dim strFriendlyName
Set objControl = ScreenItems("OLEElement1")
strFriendlyName = CreateObject("WScript.Shell").RegRead("HKCR\" &
objControl.Type & "\")
MsgBox strFriendlyName
```

Note:

In order that the example above works, a Word document should be embedded in the picture as an OLE object.

Using OLE objects

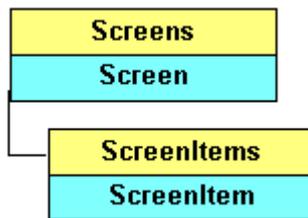
If an OLE object is used, it is possible that the properties provided by the OLE object have the same names as the general ScreenItem properties. In such cases, the ScreenItem properties have priority. The "hidden" properties of an OLE object can be accessed using the additional "object" property. Address the properties of an OLE object as follows:

OLEObjekt.object.type
Only use the form

OLEObjekt.type
In the case of identical names, the properties of the ScreenItem object are used.

Group

isplay



Object Type of the ScreenItem Object Represents the graphic object "Group Display"

Type Identifier in VBS

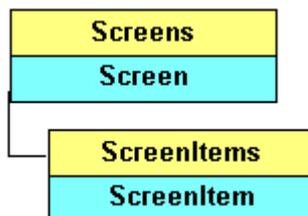
HMIGroupDisplay

Usage

In the following example, the object with the name "GroupDisplay1" is moved 10 pixels to the right:

```
'VBS44  
Dim objGroupDisplay  
Set objGroupDisplay = ScreenItems("GroupDisplay1")  
objGroupDisplay.Left = objGroupDisplay.Left + 10
```

1.2.2.9 Text List



Object Type of the ScreenItem Object Represents the graphic object "Text List"

Type Identifier in VBS

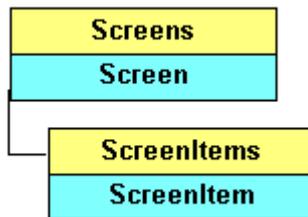
HMISymbolicIOField

Usage

In the following example, the object with the name "TextList1" is moved 10 pixels to the right:

```
'VBS45
Dim objSymIO
Set objSymIO = ScreenItems("TextList1")
objSymIO.Left = objSymIO.Left + 10
```

1.2.2.10 Status Display



Object Type of the ScreenItem Object Represents the graphic object "Status Display"

Type Identifier in VBS

HMIGraphicIOField

Usage

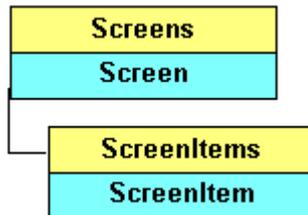
In the following example, the object with the name "StatusDisplay1" is moved 10 pixels to the right:

```
'VBS46
Dim objGraphicIO
Set objGraphicIO= ScreenItems("StatusDisplay1")
objGraphicIO.Left = objGraphicIO.Left + 10
```

1.2.3 Windows Objects

Windows objects available

1.2.3.1 Button



Object Type of the ScreenItem Object Represents the graphic object "Button"

Type Identifier in VBS

HMIButton

Usage

In the following example, the object with the name "Button1" is moved 10 pixels to the right:

```
'VBS47
Dim cmdButton
Set cmdButton = ScreenItems("Button1")
cmdButton.Left = cmdButton.Left + 10
```

Notes on Error Handling

Buttons and pushbuttons are mapped in the object model to an "HMIButton" type. Since the objects have different properties, the availability of the property (dynamic type compilation in runtime) should be queried via an exception measure. The exception measure is activated for the corresponding procedure by the following instruction:

```
On Error Resume Next
```

The instruction causes the VBScript engine to initiate the follow-on command in the case of a runtime error.

The error code can subsequently be checked using the Err object. In order to deactivate the handling of runtime errors in scripts, use the following command:

```
On Error Goto 0
```

Handling errors always relates to the procedure layer. If a script in a procedure causes an error, VBScript checks whether an error handling measure is implemented in this layer. If not, control is transferred one layer up (to the calling procedure). If there is no error handling measure here either, the control is transferred yet another layer up. This continues until either the top module level is reached or the code for runtime error handling is located. If the activation of the runtime error handling fails, the control is transferred to the top level on the internal VBScript runtime error handling. This opens an error dialog and stops the script.

The "On Error Resume Next" command can be installed on all layers (i.e. also in procedures). When the error handling measure is use, it can basically be determined whether the user is actually using the required implementation type.

In addition, it can be ensured that there is no termination of execution due to a faulty access to the object.

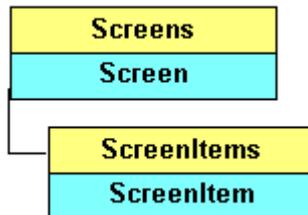
Examples of error handling

```

Sub OnClick(ByVal Item)
'VBS48
Dim objScreenItem
On Error Resume Next 'Activation of errorhandling
For Each objScreenItem In ScreenItems
If objScreenItem.Type = "HMIButton" Then
'
'=== Property "Text" available only for Standard-Button
objScreenItem.Text = "Windows"
If 0 <> Err.Number Then
HMIRuntime.Trace objScreenItem.ObjectName & ": no Windows-Button" & vbCrLf
Err.Clear 'Delete error message
End If
End If
Next
On Error Goto 0 'Deactivation of errorhandling
End Sub

```

1.2.3.2 Check Box



Object Type of the ScreenItem Object Represents the graphic object "Check Box"

Type Identifier in VBS

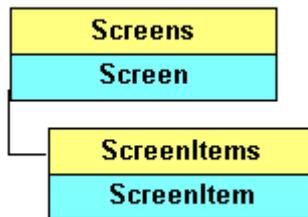
HMICheckBox

Usage

In the following example, the object with the name "CheckBox1" is moved 10 pixels to the right:

```
'VBS49
Dim chkCheckBox
Set chkCheckBox = ScreenItems("CheckBox1")
chkCheckBox.Left = chkCheckBox.Left + 10
```

1.2.3.3 Option Group



Object Type of the ScreenItem Object Represents the graphic object "Option Group"

Type Identifier in VBS

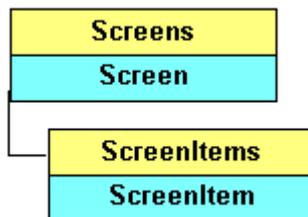
HMIOptionGroup

Usage

In the following example, the object with the name "RadioBox1" is moved 10 pixels to the right:

```
'VBS50
Dim objOptionGroup
Set objOptionGroup = ScreenItems("RadioBox1")
objOptionGroup.Left = objOptionGroup.Left + 10
```

1.2.3.4 Round Button



Object Type of the ScreenItem Object Represents the graphic object "Round Button"

Type Identifier in VBS

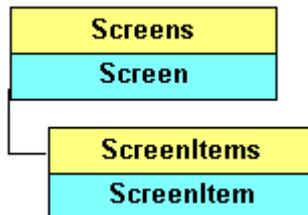
HMISwitch

Usage

In the following example, the object with the name "RoundButton1" is moved 10 pixels to the right:

```
'VBS51
Dim objSwitch
Set objSwitch= ScreenItems("RoundButton1")
objSwitch.Left = objSwitch.Left + 10
```

1.2.3.5 Slider



Object Type of the ScreenItem Object Represents the graphic object "Slider"

Type Identifier in VBS

HMISlider

Usage

In the following example, the object with the name "Slider1" is moved 10 pixels to the right:

```
'VBS53
Dim sldSlider
Set sldSlider = ScreenItems("Slider1")
sldSlider.Left = sldSlider.Left + 10
```

Notes on Error Handling

Sliders and WinCC slider controls are mapped in the object model to an "HMISlider" type. Since the objects have different properties, the availability of the property (dynamic type compilation in runtime) should be queried via an exception measure. The exception measure is activated for the corresponding procedure by the following instruction:

On Error Resume Next

The instruction causes the VBScript engine to initiate the follow-on command in the case of a runtime error.

The error code can subsequently be checked using the Err object. In order to deactivate the handling of runtime errors in scripts, use the following command:

On Error Goto 0

Handling errors always relates to the procedure layer. If a script in a procedure causes an error, VBScript checks whether an error handling measure is implemented in this layer. If not, control is transferred one layer up (to the calling procedure). If there is no error handling measure here either, the control is transferred yet another layer up. This continues until either the top module level is reached or the code for runtime error handling is located. If the activation of the runtime error handling fails, the control is transferred to the top level on the internal VBScript runtime error handling. This opens an error dialog and stops the script.

The "On Error Resume Next" command can be installed on all layers (i.e. also in procedures). When the error handling measure is use, it can basically be determined whether the user is actually using the required implementation type.

In addition, it can be ensured that there is no termination of execution due to a faulty access to the object.

Examples of error handling

```
Sub OnClick(Byval Item)
Dim ScreenItem
' Aktivierung der Fehlerbehandlung:
On Error Resume Next
For Each ScreenItem In ScreenItems
If ScreenItem.Type = "HMISlider" Then
'=== Eigenschaft "BevelColorUp" ist nur bei einem WinCC Slider Control
vorhanden
ScreenItem.BevelColorUp = 1
If (Err.Number <> 0) Then
HMIRuntime.Trace(ScreenItem.ObjectName + ": kein Windows-Slider" + vbCrLf)
' Fehlermeldung löschen
Err.Clear
End If
'=== Eigenschaft "BorderStyle" ist nur bei einem Windows-Slider vorhanden
ScreenItem.BorderStyle = 1
If (Err.Number <> 0) Then
HMIRuntime.Trace(ScreenItem.ObjectName + ": kein WinCC Slider Control" +
vbCrLf)
Err.Clear
End If
End If
```

```

Next
On Error GoTo 0 ' Deaktivierung der Fehlerbehandlung
End Sub

```

1.2.4 Controls

Special features with controls

In the case of non-WinCC controls, the version-independent ProgID is returned as the type.

It is possible to determine the version-dependent ProgID or "User friendly Name" from the ProgID: In the following example, "Control1" is a control embedded in the picture which already returns the version-independent ProgID as a result of the Type property.

Determine the version-dependent ProgID as follows:

```

dim Control, CurrentVersion
set Control = ScreenItems("Control1")
CurrentVersion = CreateObject("WScript.Shell").RegRead("HKCR\ " & Control.Type
&"\CurVer")

```

Determine the User Friendly Name as follows:

```

dim Control, FriendlyName
set Control = ScreenItems("Control1")
FriendlyName = CreateObject("WScript.Shell").RegRead("HKCR\ " & Control.Type
&"\")

```

Restrictions of VBS for Dynamization by Controls

If Controls are to be dynamized with, the following conditions must be fulfilled:

Methods

The "ByRef" declaration may only be implemented as a "Variant" (ByRef xxx as Variant)

The "ByVal" declaration may only be implemented with tag types (ByVal xxx as Long)

The "ByRef" declaration may only be implemented as a "Variant" (ByRef xxx as Variant)

The "ByVal" declaration may only be implemented with tag types (ByVal xxx as Long)

Events

The "ByRef" declaration is not permitted.

The "ByVal" declaration may only be implemented as a "Variant" (ByVal xxx as Variant)

Arrays

If arrays are used, they must be declared with (ByRef xxx As Variant).

Use of Controls from External Suppliers

If a non-WinCC control is used, it is possible that the properties provided by the control have the same names as the general ScreenItem properties. In such cases, the ScreenItem properties have priority. The "hidden" properties of an external control supplier can be accessed using the additional "object" property. Address the properties of an external control supplier as follows:

Control.object.type

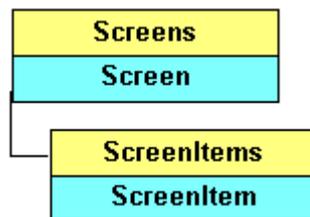
If the following form is used:

Control.type

The properties of the ScreenItem object are used in the case of identical names.

WinCC controls available

1.2.4.1 WinCC Alarm Control



Object Type of the ScreenItem Object Represents the graphic object "WinCC Alarm Control"

Type Identifier in VBS

HMIMessageView

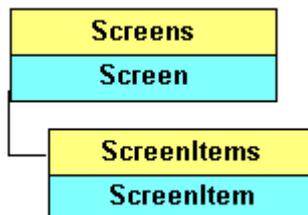
Usage

In the following example, the object with the name "Control1" is moved 10 pixels to the right:

```
'VBS54  
Dim objControl  
Set objControl = ScreenItems("Control1")  
objControl.Left = objControl.Left + 10
```

This object type has the following properties:

1.2.4.2 WinCC Digital Analog Clock



Object Type of the ScreenItem Object Represents the graphic object "WinCC Digital/Analog Clock"

Type Identifier in VBS

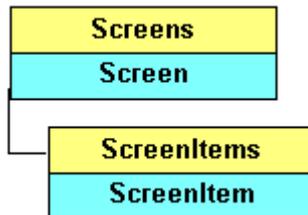
HMIClock

Usage

In the following example, the object with the name "Control1" is moved 11 pixels to the right:

```
'VBS55  
Dim objControl  
Set objControl = ScreenItems("Control1")  
objControl.Left = objControl.Left + 11
```

1.2.4.3 WinCC DXF Control



Object Type of the ScreenItem Object Represents the graphic object "WinCC DXF Control"

Type Identifier in VBS

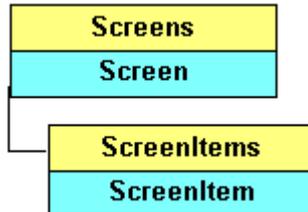
HMIDXView

Usage

In the following example, the object with the name "Control1" is moved 12 pixels to the right:

```
'VBS56  
Dim objControl  
Set objControl = ScreenItems("Control1")  
objControl.Left = objControl.Left + 12
```

1.2.4.4 WinCC Function Trend Control



Object Type of the ScreenItem Object Represents the graphic object "WinCC Function Trend Control"

Type Identifier in VBS

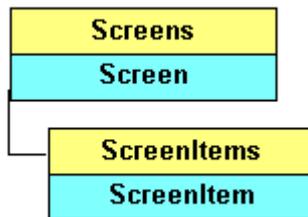
HMIFunctionTrendView

Usage

In the following example, the object with the name "Control1" is moved 13 pixels to the right:

```
'VBS57  
Dim objControl  
Set objControl = ScreenItems("Control1")  
objControl.Left = objControl.Left + 13
```

1.2.4.5 WinCC Gauge Control



Object Type of the ScreenItem Object Represents the graphic object "WinCC Gauge Control"

Type Identifier in VBS

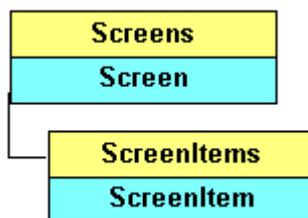
HMI Gauge

Usage

In the following example, the object with the name "Control1" is moved 14 pixels to the right:

```
'VBS58
Dim objControl
Set objControl = ScreenItems("Control1")
objControl.Left = objControl.Left + 14
```

1.2.4.6 WinCC Online Table Control



Object Type of the ScreenItem Object Represents the graphic object "WinCC Online Table Control"

Type Identifier in VBS

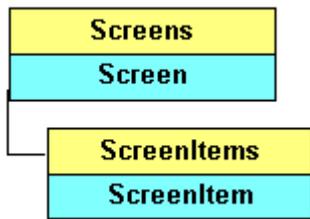
HMITableView

Usage

In the following example, the object with the name "Control1" is moved 15 pixels to the right:

```
'VBS59
Dim objControl
Set objControl = ScreenItems("Control1")
objControl.Left = objControl.Left + 15
```

1.2.4.7 WinCC Online Trend Control



Object Type of the ScreenItem Object Represents the graphic object "WinCC Online Trend Control"

Type Identifier in VBS

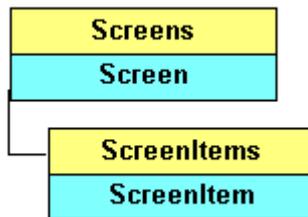
HMITrendView

Usage

In the following example, the object with the name "Control1" is moved 16 pixels to the right:

```
'VBS60
Dim objControl
Set objControl = ScreenItems("Control1")
objControl.Left = objControl.Left + 16
```

1.2.4.8 WinCC Push Button Control



Object Type of the ScreenItem Object Represents the graphic object "WinCC Push Button Control"

Type Identifier in VBS

HMIButton

Usage

In the following example, the object with the name "Control1" is moved 17 pixels to the right:

```
'VBS61
Dim objControl
Set objControl = ScreenItems("Control1")
objControl.Left = objControl.Left + 17
```

Note:

The events KeyDown, KeyUp and KeyPress cannot be addressed by VBS. If it is required to make controls dynamic with the help of VBS, no parameter must be declared with ByRef.

Notes on Error Handling

Buttons and pushbuttons are mapped in the object model to an "HMIButton" type. Since the objects have different properties, the availability of the property (dynamic type compilation in runtime) should be queried via an exception measure. The exception measure is activated for the corresponding procedure by the following instruction:

```
On Error Resume Next
```

The instruction causes the VBScript engine to initiate the follow-on command in the case of a runtime error.

The error code can subsequently be checked using the Err object. In order to deactivate the handling of runtime errors in scripts, use the following command:

```
On Error Goto 0
```

Handling errors always relates to the procedure layer. If a script in a procedure causes an error, VBScript checks whether an error handling measure is implemented in this layer. If not, control is transferred one layer up (to the calling procedure). If there is no error handling measure here either, the control is transferred yet another layer up. This continues until either the top module level is reached or the code for runtime error handling is located. If the activation of the runtime error handling fails, the control is transferred to the top level on the internal VBScript runtime error handling. This opens an error dialog and stops the script.

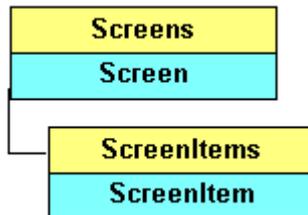
The "On Error Resume Next" command can be installed on all layers (i.e. also in procedures). When the error handling measure is use, it can basically be determined whether the user is actually using the required implementation type.

In addition, it can be ensured that there is no termination of execution due to a faulty access to the object.

Examples of error handling

```
'VBS62
Dim objScreenItem
On Error Resume Next 'Activation of errorhandling
For Each objScreenItem In ScreenItems
If objScreenItem.Type = "HMIButton" Then
'
'=== Property "Text" available only for Standard-Button
objScreenItem.Text = "Windows"
If 0 <> Err.Number Then
HMIRuntime.Trace objScreenItem.ObjectName & ": no Windows-Button" & vbCrLf
Err.Clear 'Delete error message
End If
'
'=== Property "Caption" available only for PushButton
objScreenItem.Caption = "Push"
If 0 <> Err.Number Then
HMIRuntime.Trace objScreenItem.ObjectName & ": no Control" & vbCrLf
Err.Clear
End If
End If
Next
On Error Goto 0 'Deactivation of errorhandling
```

1.2.4.9 WinCC Slider Control



Object Type of the ScreenItem Object Represents the graphic object "WinCC Slider Control"

Type Identifier in VBS

HMI Slider

Usage

In the following example, the object with the name "Control1" is moved 19 pixels to the right:

```
'VBS63
Dim objControl
Set objControl = ScreenItems("Control1")
objControl.Left = objControl.Left + 19
```

Notes on Error Handling

Sliders and WinCC slider controls are mapped in the object model to an "HMI Slider" type. Since the objects have different properties, the availability of the property (dynamic type compilation in runtime) should be queried via an exception measure. The exception measure is activated for the corresponding procedure by the following instruction:

On Error Resume Next

The instruction causes the VBScript engine to initiate the follow-on command in the case of a runtime error.

The error code can subsequently be checked using the Err object. In order to deactivate the handling of runtime errors in scripts, use the following command:

On Error Goto 0

Handling errors always relates to the procedure layer. If a script in a procedure causes an error, VBScript checks whether an error handling measure is implemented in this layer. If not, control is transferred one layer up (to the calling procedure). If there is no error handling measure here either, the control is transferred yet another layer up. This continues until either the top module level is reached or the code for runtime error handling is located. If the activation of the runtime error handling fails, the control is transferred to the top level on the internal VBScript runtime error handling. This opens an error dialog and stops the script.

The "On Error Resume Next" command can be installed on all layers (i.e. also in procedures). When the error handling measure is used, it can basically be determined whether the user is actually using the required implementation type.

In addition, it can be ensured that there is no termination of execution due to a faulty access to the object.

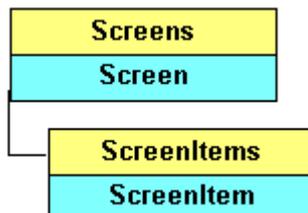
Examples of error handling

```

Sub OnClick(Byval Item)
Dim ScreenItem
' Aktivierung der Fehlerbehandlung:
On Error Resume Next
For Each ScreenItem In ScreenItems
If ScreenItem.Type = "HMISlider" Then
'=== Eigenschaft "BevelColorUp" ist nur bei einem WinCC Slider Control
vorhanden
ScreenItem.BevelColorUp = 1
If (Err.Number <> 0) Then
HMIRuntime.Trace(ScreenItem.ObjectName + ": kein Windows-Slider" + vbCrLf)
' Fehlermeldung löschen
Err.Clear
End If
'=== Eigenschaft "BorderStyle" ist nur bei einem Windows-Slider vorhanden
ScreenItem.BorderStyle = 1
If (Err.Number <> 0) Then
HMIRuntime.Trace(ScreenItem.ObjectName + ": kein WinCC Slider Control" +
vbCrLf)
Err.Clear
End If
End If
Next
On Error GoTo 0 ' Deaktivierung der Fehlerbehandlung
End Sub

```

1.2.4.10 HMI Symbol Library 1.3



Object Type of the ScreenItem Object Represents the graphic object "HMI Symbol Library "

Type Identifier in VBS

HMISymbolLibrary

Usage

In the following example, the object with the name "Control1" is moved 20 pixels to the right:

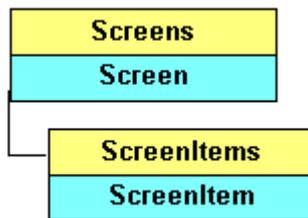
```

'VBS64
Dim objControl
Set objControl = ScreenItems("Control1")
objControl.Left = objControl.Left + 20

```

This object type has the following properties:

1.2.5 Customized object



Object Type of the ScreenItem Object Represents the graphic object "Customized Object"

Type Identifier in VBS

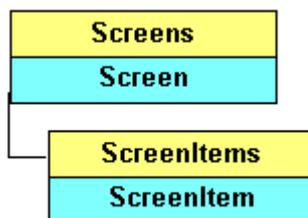
HMIScreenModule

Usage

In the following example, the object with the name "Customized object1" is moved 10 pixels to the right:

```
'VBS65
Dim objCustomObject
Set objCustomObject = ScreenItems("CustomizedObject1")
objCustomObject.Left = objCustomObject.Left + 10
```

1.2.6 Group object



Object Type of the ScreenItem Object Represents the graphic object "Group"

Type Identifier in VBS

HMIGroup

Usage

In the following example, the object with the name "Group1" is moved 10 pixels to the right:

```
'VBS66
Dim objGroup
Set objGroup = ScreenItems("Group1")
objGroup.Left = objGroup.Left + 10
```

1.3 Characteristics

Overview

The properties of the individual objects can be used to modify specific graphic objects and tags in runtime , e.g. activating an operating element per mouse click or triggering a color change by modifying a tag value.

Properties on graphic objects can be addressed via the following syntax:

```
Dim obj
Set obj = ScreenItems("object1")
obj.property = Wert
```

In the following example, the object with the name "Control1" is moved 10 pixels to the right:

```
Dim obj
Set obj = ScreenItems("control1")
obj.Left = obj.Left + 10
```

All the properties of the object and graphic object types in VBS are listed below in alphabetical order.

1.3.1 A

1.3.1.1 AccessPath Property

Displays the storage path (with hierarchy information) of a screen object (picture). The property corresponds to the full access code on the Screens Collections.

STRING (read only)

Example

In the following example, the path of the picture "ScreenWindow1" is issued:

```
'VBS67
Dim objScreen
Set objScreen = HMIRuntime.Screens("ScreenWindow1")
MsgBox objScreen.AccessPath
```

1.3.1.2 Activate Property

The data to be displayed is only requested from the archive server when this attribute is set. In order to reduce the picture opening times, this attribute should not be set and the value only dynamically changed when necessary.

Write/Read access

1.3.1.3 ActiveScreen Property

Reads out the name of the picture which contains the object which currently has the focus.

Screen (read only)

Example

The following example assigns the "strScreenName" tag to the name of the picture and displays it:

```
'VBS68
Dim strScreenName
strScreenName = HMIRuntime.ActiveScreen.ObjectName
MsgBox strScreenName
```

1.3.1.4 ActiveScreenItem Property

References the object which has the focus.

ScreenItem (read only)

Example

The following example displays the name of the object in the "ScreenWindow1" picture which has the focus:

```
'VBS69
Dim objScreen
Set objScreen = HMIRuntime.Screens("ScreenWindow1")
```

MsgBox objScreen.ActiveScreenItem.ObjectName
1.3.1.5 Actualize Property

The "Index" property references a column pair or a trend. "Actualize" defines whether a static or dynamic representation should be used for this column pair/trend.

- 0: Static display
- -1: Dynamic display

1.3.1.6 ActualPointLeft Property

Defines or returns the x-coordinate of the current corner point in relation to the original picture (top left). Each corner point is identified by an index which is derived from the number ("PointCount") of corner point available.

A change of the value can affect the properties "Width" (object width) and "Left" (x-coordinate of the object position).

1.3.1.7 ActualPointTop Property

Defines or returns the y-coordinate of the current corner point in relation to the original picture (top left). Each corner point is identified by an index which is derived from the number ("PointCount") of corner point available.

A change of the value can affect the properties "Height" (object height) and "Top" (y-coordinate of the position).

1.3.1.8 AdaptBorder Property

TRUE, when the border should be dynamically adjusted to the size of the text.
BOOLEAN write-read access.

For text list and I/O field: Read only access.

1.3.1.9 AdaptPicture Property

Defines whether the picture displayed in a picture window should be adapted to the size of the picture window in runtime or not. Read only access.

TRUE, when the picture adapts to the picture window size.

FALSE, when the picture does not adapt to the picture window size.

1.3.1.10 AdaptSize Property

Defines whether the picture window should adapt to the size of the picture displayed in it during runtime or not. Read only access.

TRUE, when the picture window adapts to the picture size.

FALSE, when the picture window does not adapt to the picture size.

1.3.1.11 AlarmHigh Property

Defines the top limit value at which an alarm should be triggered or returned.

The type of the evaluation (in percent or absolute) is defined in the "TypeAlarmHigh" property.

The "CheckAlarmHigh" property defines whether the monitoring function for the limit value is activated.

1.3.1.12 AlarmLow Property

Defines the bottom limit value at which an alarm should be triggered or returned.

The type of the evaluation (in percent or absolute) is defined in the "TypeAlarmLow" property.

The "CheckAlarmLow" property defines whether the monitoring function for the limit value is activated.

1.3.1.13 Alignment Property

Defines or returns the representation of the scale (left/right or top/bottom) according to the position of the bar graph object. The "Scaling" property must be set to TRUE for the scale to be displayed.

1.3.1.14 AlignmentLeft Property

Defines or returns the horizontal alignment of the text. Value range from 0 to 2.

0 = left

1 = centered

2 = right

1.3.1.15 AlignmentTop Property

Defines or returns the vertical alignment of the text. Value range from 0 to 2.

0 = top

1 = centered

2 = bottom

1.3.1.16 AllowPersistence Property

TRUE, when settings regarding persistence are possible. BOOLEAN write-read access.

1.3.1.17 AllServer Property

Defines that the data to be displayed in the message window is required by all servers participating in a distributed system on which Alarm Logging is activated. Write/Read access.

1.3.1.18 Analog Property

TRUE, when the clock is to be displayed as an analog clock. BOOLEAN write-read access.

1.3.1.19 AngleAlpha Property

Defines or returns depth angle a for the 3D-effect of the "3DBarGraph" object. Value range in degrees from 0 to 90.

1.3.1.20 AngleBeta Property

Defines or returns depth angle b for the 3D-effect of the "3DBarGraph" object. Value range in degrees from 0 to 90.

1.3.1.21 AngleMax Property

Defines or returns the angle on the scale at which the scale graduation ends. LONG write-read access.

The start and end of the scale graduation are described by the attributes "AngleMin" and "AngleMax" in angular degrees. AngleMin < AngleMax applies.

Angle 0 degrees is at the right side of the horizontal diameter of the graduated scale disk. Positive angle values are counted in a counterclockwise direction.

1.3.1.22 AngleMin Property

Defines or returns the angle on the scale at which the scale graduation begins. LONG write-read access.

1.3.1.23 Application Property

Returns the Graphics Designer application when the application property is used without an object identifier. If the application property is used with object identifier, it returns an application object which displays the application with which the defined object was created. Read only access.

1.3.1.24 Archive Property

The "Index" property references a pair of columns. "Archive" defines process archive values linked to the column pair. The name of the process value archive is specified in the following form: Server name::Archive name

1.3.1.25 Assignments Property

A list which contains the assignments between the output values and the actual output texts to be output.

The assignments are dependent on the list type set. The list type is defined with the ListType property.

Read only access.

1.3.1.26 AssumeOnExit Property

TRUE, when the text entered when the text should be accepted on exiting from the input field (e.g. by pressing the TAB key or clicking the mouse). BOOLEAN write-read access.

1.3.1.27 AssumeOnFull Property

TRUE, when the content of the input field is full (specified number of characters have been entered) and should be exited automatically and the input accepted. BOOLEAN write-read access.

1.3.1.28 Autorange Property

TRUE, when the value range of the Y-axis is determined automatically or defined by using the "BeginValue" and "EndValue" attributes. BOOLEAN write-read access.

1.3.1.29 AutorangeX Property

TRUE, when the value range of the X-axis is determined automatically. FALSE, when it is determined by means of the "BeginX" and "EndX" attributes. BOOLEAN write-read access.

1.3.1.30 AutorangeY Property

TRUE, when the value range of the Y-axis is determined automatically. FALSE, when it is determined by means of the "BeginY" and "EndY" attributes. BOOLEAN write-read access.

1.3.1.31 AutoScroll Property

Defines the behavior of the message window when a new message is received. BOOLEAN write-read access.

TRUE: A newly received message is appended to the list displayed in the message window and is automatically selected. The visible area of the message window is moved, if necessary.

FALSE: A newly received message is not selected. The visible area of the message window is not changed.

The targeted selection of messages is only possible when "AutoScroll" is not active.

The "AutoScroll" property is deactivated when the attribute "MsgCtrlFlag" = "-1" is set. This means that the current message in the message window appears at the top.

1.3.1.32 AutoSize Property

Defines or returns the size adaptation of the object. The following values can be set:

- 0: No size adaptation.
- 1: The picture ("PictureSelected", "PictureUnselected" properties) is adapted to the button.
- 2: The button is adapted to the picture ("PictureSelected", "PictureUnselected" properties).

1.3.1.33 Average Property

TRUE, when the average value from the last 15 values should be calculated.
BOOLEAN write-read access.

1.3.1.34 Axe Property

Defines or returns the position of the 3D bar in the coordinate system. Value range from 0 to 2.

0: The 3D-bar is displayed on the X-axis.

1: The 3D-bar is displayed on the Y-axis.

2: The 3D-bar is displayed on the Z-axis.

1.3.1.35 AxisSection Property

Defines or returns the distance between two long axis sections. The information on the distance is given in scale units and is dependent on the minimum and maximum values configured.

1.3.2 B

1.3.2.1 BackBorderWidth Property

Defines or returns the width of the 3D border in pixels. The value for the width is dependent on the size of the object.

1.3.2.2 BackColor Property

Function

Defines or returns the background color for the object.

The background color is not displayed if "transparent" is defined as the fill pattern.

LONG (write-read access)

Special features of the WinCC slider control

The background color only takes effect when the object is at least partially filled.

Determination of Color Value

The color is displayed in RGB format (Red, Green, Blue). Enter the appropriate decimal value for each of the three RGB values.

Example:

RGB(200, 150, 100)

Example

The following example defines the background of the "ScreenWindow1" picture to red:

```
'VBS70
Dim objScreen
Set objScreen = HMIRuntime.Screens("ScreenWindow1")
objScreen.BackColor = RGB(255, 0, 0)
```

1.3.2.3 BackColor2 Property

Defines or returns the bar color for the display of the current value. LONG write-read access.

1.3.2.4 BackColor3 Property

Defines or returns the color of the bar background. LONG write-read access.

1.3.2.5 BackColorBottom Property

Defines or returns the color for the bottom/right part of the slider. LONG write-read access.

1.3.2.6 BackColorTop Property

Defines or returns the color for the top/left part of the slider. LONG write-read access.

1.3.2.7 **BackFlashColorOff Property**

Defines or returns the color of the object background for the flash status "Off".
LONG write-read access.

1.3.2.8 **BackFlashColorOn Property**

Defines or returns the color of the object background for the flash status "On".
LONG write-read access.

1.3.2.9 **Background Property**

TRUE, when the background of the 3D-bar graph object should be visible.
BOOLEAN write-read access.

1.3.2.10 **BackgroundPicture Property**

Returns the picture name of the background picture for the graduated scale disk.
Read only access

1.3.2.11 **BackStyle Property**

WinCC Digital/Analog Clock

Defines the type of background of the analog clock:

- 0: The rectangular background of the clock is filled by the specified background color.
- 1: The round numbered face of the clock is filled by the specified background color. This enables a round analog clock to be displayed.
- 2: Numbered face and rectangular background are transparent.

WinCC Gauge Control

Defines the type of background of the gauge:

- 0: The rectangular or square background of the gauge has a border color is filled with the specified color. The circular graduated scale disk is filled by the specified background color.
- 1: The rectangular or square background of the gauge is transparent. The circular graduated scale disk is filled by the specified background color. This enables a circular gauge to be displayed.
- 2: The rectangular or square background and graduated scale disk are transparent.

WinCC Slider Control

Defines whether the object background should be transparent.

- 0: The object background is not transparent
- 1: The object background is transparent

HMI Symbol Library 1.3

Defines the icon background transparency. Write/Read access.

- 0: The background is transparent and, thus, invisible.
- 1: The background is visible, the color of the background is defined by the "Background Color" attribute.

1.3.2.12 BarBackColor Property

Defines the background color in the area of the slider. The area stretches from "RangeMin" to "RangeMax".

1.3.2.13 BarDepth Property

Defines or returns the depth of the bar in pixels.

1.3.2.14 BarFillColor Property

Defines the fill color in the area of the slider. The area stretches from "RangeMin" to the position of the slider.

1.3.2.15 BarHeight Property

Defines or returns the height of the bar in pixels.

1.3.2.16 BarWidth Property

Defines or returns the width of the bar in pixels.

1.3.2.17 BasePicReferenced Property

TRUE, when the picture assigned in the object status display should be saved. Otherwise, only the associated object reference is saved. Read only access.

1.3.2.18 BasePicTransColor Property

Defines or returns which color of the assigned bitmap object (.bmp, .dib) should be set to "transparent". LONG write-read access.

The color is then only set to "transparent" when the "BasePicUseTransColor" property has the value TRUE.

1.3.2.19 BasePicture Property

Returns the basic picture for the object status display. Read only access.

The picture (*.BMP or *.DIB) must be located in the "GraCS" directory of the current project so that it can be integrated.

In this context, the "BasePicReferenced" property defines whether the basic picture should be saved together with the object status display or referenced.

1.3.2.20 BasePicUseTransColor Property

TRUE, when the configured color ("BasePicTransparentColor" property) of the bitmap objects should be set to "transparent". BOOLEAN write-read access.

1.3.2.21 BaseScreenName Property

Function

Defines or returns the current basic picture.

STRING (write-read access)

A picture change is executed using the

HMIRuntime.BaseScreenName = (<Serverpräfix>::)<Neues Grundbild>
command.

When reading out the "BaseScreenName" property, only the picture name without server prefix is returned.

Note:

Always enter picture names without the extension "PDL" for reasons of compatibility with future versions.

Example

The following example executes a picture change to "bild1.pdl":

```
HMIRuntime.BaseScreenName = "bild1"
```

1.3.2.22 BaseY Property

Defines or returns the vertical distance of the bottom bar edge to the top edge of the object field.

1.3.2.23 BaseX Property

Defines or returns the horizontal distance of the right bar edge to the left edge of the object field in pixels.

1.3.2.24 BeginTime Property

WinCC Online Table Control

The "Index" property references a pair of columns. "BeginTime" defines the start time for displaying this column pair. Write/Read access.

WinCC Online Trend Control

The "Index" property references a trend. "BeginTime" defines the start time for displaying this trend. Whether the information is evaluated is dependent on the "TimeRange" and "CommonX" properties.

1.3.2.25 BeginValue Property

The "Index" property references a trend. "BeginValue" defines the lower limit of the value range to be displayed for the trend. Whether the information is evaluated is dependent on the "Autorange" and "CommonY" properties.

1.3.2.26 BeginX Property

Defines or returns the lower limit of the X-axis of a trend referenced with the "Index" property. Whether the information is evaluated is dependent on the "AutorangeX" and "CommonX" properties.

1.3.2.27 BeginY Property

Defines or returns the lower limit of the Y-axis of a trend referenced with the "Index" property. Whether the information is evaluated is dependent on the "AutorangeY" and "CommonY" properties.

1.3.2.28 BevelColorDown Property

Defines the color of the following border sections in the case of 3D representation of the borders:

- with depressed bevel ("BevelStyle" = 1): top and left bevel section
- with raised bevel ("BevelStyle" = 2): bottom and right bevel section

1.3.2.29 BevelColorUp Property

Defines the color of the following border sections in the case of 3D representation of the borders:

- with depressed bevel ("BevelStyle" = 1): bottom and right bevel section
- with raised bevel ("BevelStyle" = 2): top and left bevel section

1.3.2.30 BevelInner Property

Defines or returns the appearance of the inner part of the object bevel. Write/Read access.

- 0: inner part not available
- 1: "depressed" appearance
- 2: "raised" appearance
- 3: uniform gray border
- 4 or higher: uniformly colored order, border color = background color

1.3.2.31 BevelOuter Property

Defines or returns the appearance of the outer part of the object bevel. Write/Read access.

- 0: inner part not available
- 1: "depressed" appearance
- 2: "raised" appearance
- 3: uniform gray border
- 4 or higher: uniformly colored order, border color = background color

1.3.2.32 BevelWidth Property

Defines or returns the border width for the inner part of the border (inner bevel) and for the outer border part (outer bevel) in pixels. Write/Read access.

1.3.2.33 BitNumber Property

Defines or returns the bit whose status must change in order to trigger a change of value. The tag used must be of the type BYTE, WORD or DWORD.

1.3.2.34 BlinkColor Property

Defines the color of the icon in the flash picture. LONG write-read access.

1.3.2.35 BlinkMode Property

Defines the appearance of the icon surface of the flash picture.

- 0: Flashing is switched off.
- 1: The flash picture is invisible.
- 2: The flash picture is assigned a colored, shaded surface. The color of the surface corresponds to the setting of the "BlinkColor" property.
- Solid - 3 (3): The flash picture is assigned a colored, unshaded surface. The color of the surface corresponds to the setting of the "BlinkColor" property.

1.3.2.36 BlinkSpeed Property

Defines the length of the flashing interval.

- Fast - 250: The length of the flashing interval is 250 ms.
- Medium - 500: The length of the flashing interval is 500 ms.
- Slow - 1000: The length of the flashing interval is 1000 ms.

The default value is Medium - 500.

1.3.2.37 BorderBackColor Property

Defines or returns the background color of the line for the object. LONG write-read access. The background color is only visible with the property setting "BorderWidth" > 0.

1.3.2.38 BorderColor Property

Defines or returns the line color for the object. LONG write-read access.

1.3.2.39 BorderColorBottom Property

Defines or returns the border color for the bottom/right part of the object. LONG write-read access.

1.3.2.40 BorderColorTop Property

Defines or returns the border color for the top/left part of the object. LONG write-read access.

1.3.2.41 BorderEndStyle Property

Defines or returns the line end style of the object. LONG write-read access.

1.3.2.42 BorderFlashColorOff Property

Defines or returns the color of the object lines for the flashing status "Off". LONG write-read access.

1.3.2.43 BorderFlashColorOn Property

Defines or returns the color of the object lines for the flashing status "On". LONG write-read access.

1.3.2.44 BorderStyle Property

Defines or returns the line style for the object. Value range from 0 to 4.

0 = solid line

1 = dashed line

2 = dotted line

3 = dash-dotted line

4 = dash-dot-dot line

1.3.2.45 **BorderWidth Property**

Defines or returns the line weight (in pixels) for the object.

WinCC Gauge Control:

Defines or returns the width of the middle border part in pixels.

The object border is composed of three parts. The middle part of the object border is described by the "BorderWidth" property.

The color of the middle border part is in the background color.

1.3.2.46 **BottomConnectedConnectionPointIndex Property**

Specifies or sets the index number of the bottom connecting point.

LONG write-read access.

1.3.2.47 **BottomConnectedObjectName Property**

Specifies or sets the object name of the object which is docked on at the bottom connecting point.

LONG write-read access.

1.3.2.48 **BoxAlignment Property**

TRUE, when the fields are arranged aligned to the right. BOOLEAN write-read access.

1.3.2.49 **BoxCount Property**

Defines or returns the number of fields. Value range from 0 to 31.

1.3.2.50 **BoxType Property**

Defines or returns the field type. Value range from 0 to 2.

- 0: Output
- 1: Input
- 2: I/O field

1.3.2.51 ButtonColor Property

Defines or returns the color of the slider. LONG write-read access.

1.3.2.52 ButtonCommand Property

On changing the value of ButtonCommand, a message is issued to the WinCC Alarm Control in order to influence the display in the message window.

Value (hex); Value (dec); Called function:

- 0x00000001; 1; Process Message window
- 0x00000002; 2; or 0x00000004; 4; Archive
- 0x00200000; 2097152; Display Lock window
- 0x00000008; 8; Acknowledge Central Signaling Device
- 0x00000010; 16; Single acknowledgment
- 0x00000020; 32; Group acknowledgement
- 0x00000040; 64; Auto-Scroll on/off
- 0x00000080; 128; Selection
- 0x00000100; 256; Set lock
- 0x00000200; 512; Protocol functions
- 0x00000800; 2048; Emergency quit (reset)
- 0x00001000; 4096; Start of list
- 0x00002000; 8192; End of list
- 0x00004000; 16384; Next message
- 0x00008000; 32768; Previous message
- 0x00010000; 65536; Info text
- 0x00020000; 131072; Comments (archive)
- 0x00040000; 262144; Loop-in-alarm
- 0x00100000; 1048576; Print
- 0x00400000; 4194304; Lock/Release message
- 0x00800000; 8388608; Sort
- 0x01000000; 16777216; Dialog for time synchronization

1.3.2.53 Button1Width Property

Defines or returns the width of the Button 1 in pixels.

When the SameSize property is set to TRUE, all the buttons are specified the same width.

1.3.2.54 Button2Width Property

Defines or returns the width of the Button 2 in pixels.

When the SameSize property is set to TRUE, all the buttons are specified the same width.

1.3.2.55 Button3Width Property

Defines or returns the width of the Button 3 in pixels.

When the SameSize property is set to TRUE, all the buttons are specified the same width.

1.3.2.56 Button4Width Property

Defines or returns the width of the Button 4 in pixels.

When the SameSize property is set to TRUE, all the buttons are specified the same width.

1.3.3 C

1.3.3.1 Caption Property

Application and picture windows

TRUE, when the application or picture window has a title bar in runtime. Read only access.

The Caption property must be set to TRUE when the application or picture window should have Maximize and Close buttons.

Controls

Defines or returns the text to be displayed on the label on the button or in the title bar (Online Trend Control and Online Table Control). Write/Read access.

1.3.3.2 CaptionColor Property

Defines or returns the color of the element labeling. LONG write-read access.

1.3.3.3 CaptionFont Property

Returns the values for font, font style and font size as well as the "Underline" and "Strikethrough" effects for the element labeling. Read only access.

1.3.3.4 CaptionOffset Property

Defines or returns the distance of the element labeling in relation to the top edge of the object. The element labeling can only be positioned along the vertical diameter of the graduated scale disk. The value of the attribute is related to the height of the object and is measured from the top edge of the object to the base of the text. Write/Read access.

The value range is 0 to 1:

0: The base of the text is at the top limit of the object. The text is no longer visible because it is outside the object.

1: The base of the text is at the bottom limit of the object.

1.3.3.5 CaptionText Property

Defines or returns the window title which is displayed in runtime. The Caption property must be set to TRUE.

1.3.3.6 CellCut Property

TRUE, when the content of the cells in a message line should be cut if the column width is too small. BOOLEAN write-read access.

1.3.3.7 CenterColor Property

Defines or returns the color of the circular center of the scale (cover of the pointer axis). LONG write-read access.

1.3.3.8 CenterScale Property

Defines or returns the diameter of the circular center of the scale (cover of the pointer axis) in relation to the smaller value of the geometric width and height attributes. Write/Read access.

The value range is 0.03 to 1:

1: The diameter corresponds to the smaller value of the "Width" or "Height" geometric values.

1.3.3.9 CheckAlarmHigh Property

TRUE, when the "AlarmHigh" limit value is to be monitored. BOOLEAN write-read access.

The limit value, the display on reaching the limit value and the type of evaluation are defined by means of the "AlarmHigh", "ColorAlarmHigh" and "TypeAlarmHigh" properties.

1.3.3.10 CheckAlarmLow Property

TRUE, when the "AlarmLow" limit value is to be monitored. BOOLEAN write-read access.

The limit value, the display on reaching the limit value and the type of evaluation are defined by means of the "AlarmLow", "ColorAlarmLow" and "TypeAlarmLow" properties.

1.3.3.11 CheckLimitHigh4 Property

TRUE, when the "Reserve 4" upper limit value should be monitored. BOOLEAN write-read access.

The limit value, the display on reaching the limit value and the type of evaluation are defined by means of the "LimitHigh4", "ColorLimitHigh4" and "TypeLimitHigh4" properties.

1.3.3.12 CheckLimitHigh5 Property

TRUE, when the "Reserve 5" upper limit value should be monitored. BOOLEAN write-read access.

The limit value, the display on reaching the limit value and the type of evaluation are defined by means of the "LimitHigh5", "ColorLimitHigh5" and "TypeLimitHigh5" properties.

1.3.3.13 CheckLimitLow4 Property

TRUE, when the "Reserve 4" lower limit value should be monitored. BOOLEAN write-read access.

The limit value, the display on reaching the limit value and the type of evaluation are defined by means of the "LimitLow4", "ColorLimitLow4" and "TypeLimitLow4" properties.

1.3.3.14 CheckLimitLow5 Property

TRUE, when the "Reserve 5" lower limit value should be monitored. BOOLEAN write-read access.

The limit value, the display on reaching the limit value and the type of evaluation are defined by means of the "LimitLow5", "ColorLimitLow5" and "TypeLimitLow5" properties.

1.3.3.15 CheckToleranceHigh Property

TRUE, when the "ToleranceHigh" limit value is to be monitored. BOOLEAN write-read access.

The limit value, the display on reaching the limit value and the type of evaluation are defined by means of the "ToleranceHigh", "ColorToleranceHigh" and "TypeToleranceHigh" properties.

1.3.3.16 CheckToleranceLow Property

TRUE, when the "ToleranceLow" limit value is to be monitored. BOOLEAN write-read access.

The limit value, the display on reaching the limit value and the type of evaluation are defined by means of the "ToleranceLow", "ColorToleranceLow" and "TypeToleranceLow" properties.

1.3.3.17 CheckWarningHigh Property

TRUE, when the "WarningHigh" limit value is to be monitored. BOOLEAN write-read access.

The limit value, the display on reaching the limit value and the type of evaluation are defined by means of the "WarningHigh", "ColorWarningHigh" and "TypeWarningHigh" properties.

1.3.3.18 CheckWarningLow Property

TRUE, when the "WarningLow" limit value is to be monitored. BOOLEAN write-read access.

The limit value, the display on reaching the limit value and the type of evaluation are defined by means of the "WarningLow", "ColorWarningLow" and "TypeWarningLow" properties.

1.3.3.19 ClearOnError Property

TRUE, when the field entry is automatically deleted in the case of invalid input. BOOLEAN write-read access.

1.3.3.20 ClearOnNew Property

TRUE, when the field entry is deleted as soon as the I/O field has the focus. BOOLEAN write-read access.

1.3.3.21 Closeable Property

TRUE, when the window can be closed in runtime. BOOLEAN write-read access.

1.3.3.22 CloseButton Property

TRUE, when the window is provided with a "Close" button. Read only access.

1.3.3.23 CoarseGrid Property

TRUE when the value axis is scaled by long tick marks. The distance between two long tick marks can be changed using the "CoarseGridValue" property. BOOLEAN write-read access.

1.3.3.24 CoarseGridX Property

TRUE, when the X-axis graduation is scaled by long tick marks. The distance between two long tick marks can be changed using the "CoarseGridValueX" property. BOOLEAN write-read access.

1.3.3.25 CoarseGridY Property

TRUE, when the Y-axis graduation is scaled by long tick marks. The distance between two long tick marks can be changed using the "CoarseGridValueY" property. BOOLEAN write-read access.

1.3.3.26 CoarseGridValue Property

Defines the distance between two long tick marks in the scale. Whether the information is evaluated is dependent on the value of the "CoarseGrid" property.

1.3.3.27 CoarseGridValueX Property

Defines or returns the distance between two long tick marks on the graduation scale of the X-axis. Whether the information is evaluated is dependent on the value of the "CoarseGridX" property.

1.3.3.28 CoarseGridValueY Property

Defines or returns the distance between two long tick marks on the graduation scale of the Y-axis. Whether the information is evaluated is dependent on the value of the "CoarseGridY" property.

1.3.3.29 CollectValue Property

Contains the respective status of the active message class in runtime as the start value. LONG write-read access.

The value can be determined from the group display of hierarchically subordinate pictures by making it dynamic using a tag.

1.3.3.30 ColMove Property

TRUE, when the arrangement of columns can be changed. BOOLEAN write-read access.

1.3.3.31 Color Property

The "Index" property references a column pair or a trend. "Color" defines the color of the font in the column or the trend. LONG write-read access. The color is defines as an RGB value.

1.3.3.32 ColorAlarmHigh Property

Defines or returns the bar color for the "AlarmHigh" limit value. LONG write-read access.

The "CheckAlarmHigh" property must have been set to TRUE if the bar color should change on reaching the limit value.

1.3.3.33 ColorAlarmLow Property

Defines or returns the bar color for the "AlarmLow" limit value. LONG write-read access.

The "CheckAlarmLow" property must have been set to TRUE if the bar color should change on reaching the limit value.

1.3.3.34 ColorBottom Property

Defines or returns the color for the bottom/right stop of the slider object. LONG write-read access.

1.3.3.35 ColorChangeType Property

TRUE, if the change of color should occur segment by segment in the case of a color change (e.g. on reaching a limit value). If set to FALSE, it defines the change of color for the entire bar. BOOLEAN write-read access.

1.3.3.36 ColorLimitHigh4 Property

Defines or returns the color for the "Reserve 4" upper limit value. LONG write-read access.

The "CheckLimitHigh4" property must have been set to TRUE if the bar color should change on reaching the limit value.

1.3.3.37 ColorLimitHigh5 Property

Defines or returns the color for the "Reserve 5" upper limit value. LONG write-read access.

The "CheckAlarmHigh5" property must have been set to TRUE if the bar color should change on reaching the limit value.

1.3.3.38 ColorLimitLow4 Property

Defines or returns the color for the "Reserve 4" lower limit value. LONG write-read access.

The "CheckAlarmLow4" property must have been set to TRUE if the bar color should change on reaching the limit value.

1.3.3.39 ColorLimitLow5 Property

Defines or returns the color for the "Reserve 5" lower limit value. LONG write-read access.

The "CheckAlarmLow5" property must have been set to TRUE if the bar color should change on reaching the limit value.

1.3.3.40 ColorToleranceHigh Property

Defines or returns the color for the "ToleranceHigh" upper limit value. LONG write-read access.

The "CheckToleranceHigh" property must have been set to TRUE if the bar color should change on reaching the limit value.

1.3.3.41 ColorToleranceLow Property

Defines or returns the color for the "ToleranceLow" lower limit value. LONG write-read access.

The "CheckToleranceLow" property must have been set to TRUE if the bar color should change on reaching the limit value.

1.3.3.42 ColorTop Property

Defines or returns the color for the top/left stop of the slider object. LONG write-read access.

1.3.3.43 ColorWarningHigh Property

Defines or returns the color for the "WarningHigh" upper limit value. LONG write-read access.

The "CheckWarningHigh" property must have been set to TRUE if the bar color should change on reaching the limit value.

1.3.3.44 ColorWarningLow Property

Defines or returns the color for the "WarningLow" lower limit value. LONG write-read access.

The "CheckWarningLow" property must have been set to TRUE if the bar color should change on reaching the limit value.

1.3.3.45 ColTitle Property

TRUE, when the columns in the message window should have a title bar. BOOLEAN write-read access.

1.3.3.46 Command Property

TRUE, when updating of the values displayed in the control should be forced.

1.3.3.47 ColWidth Property

TRUE, when it should be possible to change the widths of the columns in the message window. The width of the columns can only be changed, however, when the "AutoScroll" property is not active. BOOLEAN write-read access.

1.3.3.48 CommonTime Property

TRUE, when a common time column is to be used in the table window. BOOLEAN write-read access.

1.3.3.49 CommonX Property

TRUE, when the trends in the trend window should be displayed with a common X-axis. BOOLEAN write-read access.

1.3.3.50 CommonY Property

TRUE, when the trends in the trend window should be displayed with a common Y-axis. BOOLEAN write-read access.

1.3.3.51 ContinuousChange Property

Defines the type of transfer of the value defined by the slider ("Position" property) in Runtime:

- FALSE: The value of the "Position" property is transferred when the mouse button is released.
- TRUE: The value of the "Position" property is transferred immediately following a change of the slider position.

1.3.3.52 CurrentContext Property

In the case of a picture window, the server from which the picture comes and contains the script is read out.

The "CurrentContext" property can return different results: If, for example, a picture window displaying a server picture is set in a local basic picture, distinction is made between two cases:

- The "CurrentContext" property is used in an action of the picture window picture: The result is the return of the symbolic computer name of the server (Package property) extended by two colons, e.g. "WinCCProject_MyComputer::".
- The "CurrentContext" property is used in an action of the basic picture: The result is returned in the form of an empty character string.

1.3.3.53 Cursor Property

Controls the appearance of the cursor in runtime when positioned over an icon.

- 0: The cursor appears as an arrow and does not change when positioned over the icon.
- 1: The cursor appears as a 3D arrow accompanied by a green lightning symbol. In runtime, this indicates that the object concerned can be operated.

1.3.3.54 CursorControl Property

TRUE, when Alpha Cursor mode is activated, the cursor skips to the next field in the TAB sequence after exiting the field. BOOLEAN write-read access.

To do this, the "CursorMode" property must be set to TRUE.

1.3.3.55 CurveForm Property

WinCC Function Trend Control

Defines how the measuring points of a trend referenced by the "Index" property should be connected. Write/Read access.

WinCC Online Trend Control

The "Index" property references a trend. "CurveForm" defines how the measuring points should be connected.

- 0x00000012: Representation of the measuring points.
- 0x00000014: Measuring points are connected linearly.
- 0x00000011: Measuring points are connected via a step curve.
- 0x00000021: The area under the linearly connected trend is filled.
- 0x00000022: The area under the step curve is filled.

1.3.4 D

1.3.4.1 DangerColor Property

Defines or returns the color of the danger zone on the scale. LONG write-read access.

1.3.4.2 Danger Property

Defines or returns the beginning of the "danger zone". The zone stretches from the "danger" value to the end of the scale. Write/Read access.

1.3.4.3 DataFormat Property

Returns the data type of the I/O field object. Read only access.

Value range from 0 to 3.

0: Binary

1: Decimal

2: Heximal

3: String

1.3.4.4 DataIndex Property

Returns the current index of the data of the current trend.

1.3.4.5 DataX Property

Inserts a single data record and must be set before calling "InsertData".

1.3.4.6 DataXY Property

Inserts several data records as an array with pairs of values and must be set before calling "InsertData".

The data in the array is assumed when "DataX" is of the VT_EMPTY type. Otherwise, the "InsertData" attribute used the single value pair resulting from DataX and DataY.

1.3.4.7 DataY Property

Inserts a single data record and must be set before calling "InsertData".

1.3.4.8 DeleteData Property

Deletes data in the data buffer of the current trend.

TRUE: All trend data is deleted.

FALSE: The value pair at the "DataIndex" position are deleted.

1.3.4.9 Delta Property

Defines or returns the value difference between two main scale graduation marks. Write/Read access.

1.3.4.10 DesiredCurveColor Property

Defines the color of a setpoint trend which belongs to a trend referenced by the "Index" property. The color is defines as an RGB value. Whether the information is evaluated is dependent on the value of the "DesiredCurveVisible" property.

1.3.4.11 **DesiredCurveCurveForm Property**

Defines the form of representation of a setpoint trend which belongs to a trend referenced by the "Index" property. Whether the information is evaluated is dependent on the value of the "DesiredCurveVisible" property.

0x00000011: Measuring points are connected by a solid line via a step curve

0x00000012: Representation of the measuring points

0x00000014: Measuring points are connected linearly with a solid line

0x00000021: The area under the linearly connected trend is filled.

0x00000022: The area under the step curve is filled.

0x00000031: Measuring points are connected by a dashed line via a step curve

0x00000032: Measuring points are connected linearly with a dashed line

1.3.4.12 **DesiredCurveSourceNumberOfUAValues Property**

Defines the number of value pairs of a setpoint trend which belongs to a trend referenced by the "Index" property. Whether the information is evaluated is dependent on the value of the "DesiredCurveVisible" property.

1.3.4.13 **DesiredCurveSourceUAArchive Property**

Defines the name of the user archive from which the value of a setpoint trend, which belongs to a trend referenced by "Index", is read. Whether the information is evaluated is dependent on the value of the "DesiredCurveVisible" property.

1.3.4.14 **DesiredCurveSourceUAArchiveStartID Property**

Defines the starting point for the value of a setpoint trend, which belongs to a trend referenced by "Index", from which the values should be read from the archive. Whether the information is evaluated is dependent on the value of the "DesiredCurveVisible" property.

1.3.4.15 **DesiredCurveSourceUAColumnX Property**

Defines the column in the user archive from which the X-values of a setpoint trend, which belongs to a trend referenced by "Index", should be read. Whether the information is evaluated is dependent on the value of the "DesiredCurveVisible" property.

1.3.4.16 DesiredCurveSourceUAColumnY Property

Defines the column in the user archive from which the Y-values of a setpoint trend, which belongs to a trend referenced by "Index", should be read. Whether the information is evaluated is dependent on the value of the "DesiredCurveVisible" property.

1.3.4.17 DesiredCurveVisible Property

TRUE, a setpoint trend which belongs to a trend referenced by "Index" should be displayed. BOOLEAN write-read access.

1.3.4.18 Direction Property

Defines or returns the bar direction or the position of the slider object. BOOLEAN write-read access. Value range from 0 to 3.

0 = top

1 = bottom

2 = left

3 = right

1.3.4.19 DXFFilename Property

Specifies the name of the DXF file to be displayed. The extension "DXF" can be omitted and is then automatically appended.

If the name is specified with path, the file is also copied in the GraCS directory of the project directory structure. This ensures that when copying the project, all the picture files are copied at the time. If the DXF file is entered manually and the specified path is not available (or still not), a message appears.

An attempt is initially made, in runtime, to open the file from the specified path. If this is not possible, the file in the project directory is used.

If the name is entered without the path, the file is expected in the GraCS directory of the project directory.

1.3.5 E

1.3.5.1 Edit Property

Activates Editing mode for a cell as long as the "Editable" property has been set to TRUE for the corresponding column.

1.3.5.2 Editable Property

The "Index" property references a pair of columns. "Editable" defines whether the column pair should be editable. BOOLEAN write-read access.

1.3.5.3 EditAtOnce Property

TRUE, when, after skipping to a field with the TAB key, input should be possible immediately without any further action. BOOLEAN write-read access.

1.3.5.4 Enabled Property

Function

Enables or disables possible operation of an object or issues the corresponding value. TRUE: Enable operation, FALSE: Operation is disabled.

BOOLEAN write-read access.

Example

The following example disables all objects in the picture "NewPDL1":

```
'VBS71
Dim objScreen
Dim objScrltem
Dim lngIndex
Dim strName
lngIndex = 1
Set objScreen = HMIRuntime.Screens("NewPDL1")
For lngIndex = 1 To objScreen.ScreenItems.Count
strName = objScreen.ScreenItems.Item(lngIndex).ObjectName 'Read names of
objects
Set objScrltem = objScreen.ScreenItems(strName)
objScrltem.Enabled=False 'Lock object
Next
```

1.3.5.5 EndAngle Property

Defines or returns the end of the object. The information is in counterclockwise direction in degrees, beginning at the 12:00 clock position.

1.3.5.6 EndTime Property

Online Table Control

The "Index" attribute references a pair of columns. "EndTime" defines the end time for displaying this column pair. Whether the information is evaluated is dependent on the TimeRange" and "CommonTime" properties. Write/Read access.

Online Trend Control

The "Index" attribute references a trend. "EndTime" defines the end time for displaying this trend. Whether the information is evaluated is dependent on the "Autorange", "TimeRange" and "CommonX" properties.

1.3.5.7 EndValue Property

The "Index" property references a trend. "EndValue" defines the upper limit of the value range to be displayed for the trend. Whether the information is evaluated is dependent on the "Autorange" and "CommonY" properties.

1.3.5.8 EndX Property

Defines the upper limit of the X-axis of a trend referenced with "Index". Whether the information is evaluated is dependent on the "AutorangeX" and "CommonX" properties.

1.3.5.9 EndY Property

Defines the upper limit of the Y-axis of a trend referenced with "Index". Whether the information is evaluated is dependent on the "AutorangeY" and "CommonY" properties.

1.3.5.10 Error Property

Function

Error of the "LastError" property. The error is only provided in English.

STRING (read only)

The following error messages are defined:

OKDESCR	_T(" ")
OPERATION_ERRORDESC	_T("Operation Failed")
VARIABLE_ERRORDESC	_T("Variable not found")

In order that Error returns a value, a read process must be executed beforehand.

Example

The following example displays the error for "Tag1":

```
'VBS72
Dim objTag
Set objTag = HMIRuntime.Tags("Tag1")
objtag.Read
MsgBox objTag.Error
```

1.3.5.11 Exponent Property

TRUE, when the display of numbers should be with exponents (e.g."1.00e+000").
BOOLEAN write-read access.

1.3.5.12 ExtendedOperation Property

TRUE, when the slider regulator is set at the respective end value (minimum/maximum value). This is done by clicking the mouse in an area outside the current regulator setting. BOOLEAN write-read access.

1.3.6 F

1.3.6.1 FillColor Property

Defines or returns the fill pattern color for the object.

LONG (write-read access)

Determination of Color Value

The color is displayed in RGB format (Red, Green, Blue). Enter the appropriate decimal value for each of the three RGB values.

Example:

RGB(200, 150, 100)

Example

The following example defines the fill color for "ScreenWindow1" to blue:

```
'VBS73
Dim objScreen
Set objScreen = HMIRuntime.Screens("ScreenWindow1")
objScreen.FillStyle = 131075
objScreen.FillColor = RGB(0, 0, 255)
```

1.3.6.2 Filling Property

TRUE, when the object can be filled by closed border lines (e.g. representing the fill level of a tank). BOOLEAN write-read access.

The fill level of the object is set by means of the "FillingIndex" property.

1.3.6.3 FillingIndex Property

Defines the %age value (related to the height of the object) to which the object with closed border line is to be filled.

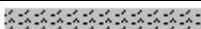
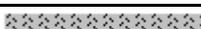
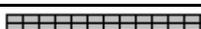
The fill level is represented by the current background color. The unfilled background is transparent.

1.3.6.4 FillStyle Property

Defines or returns the fill style for the object.

LONG (write-read access)

Fill Pattern	Value	Fill Pattern	Value
< Transparent >	65536		196623
< Solid >	0		196624
	131075		196625
	131076		196626
	131077		196627
	131074		196628
	131072		196629
	131073		196630
	196608		196631
	196609		196632
	196610		196633
	196611		196634
	196612		196635
	196613		196636
	196614		196637
	196615		196638
	196616		196639

Fill Pattern	Value	Fill Pattern	Value
	196617		196640
	196618		196641
	196619		196642
	196620		196643
	196621		196644
	196622	--	--

Example

The following example sets the fill style for "ScreenWindow1" to transparent:

```
Dim myScr
Set myScr = HMIRuntime.Screens("Bildfenster1")
myScr.FillStyle = 65536
```

1.3.6.5 FillStyle2 Property

Defines or returns the fill style of the bar.

1.3.6.6 FineGrid Property

TRUE, when the value axis is scaled by short tick marks. The distance between two short tick marks can be changed using the "FineGridValue" property.
BOOLEAN write-read access.

1.3.6.7 FineGridValue Property

Defines the distance between two short tick marks in the scale. Whether the information is evaluated is dependent on the value of the "FineGrid" property.

1.3.6.8 FineGridValueX Property

Defines the distance between two short tick marks on the X-axes scaling. Whether the information is evaluated is dependent on the value of the "FineGridX" property.

1.3.6.9 FineGridValueY Property

Defines the distance between two short tick marks on the Y-axis scaling. Whether the information is evaluated is dependent on the value of the "FineGridX" property.

1.3.6.10 FineGridX Property

TRUE, when the X-axis graduation is scaled by short tick marks. The distance between two short tick marks can be changed using the "FineGridValueX" property.

1.3.6.11 FineGridY Property

TRUE, when the Y-axis graduation is scaled by short tick marks. The distance between two short tick marks can be changed using the "FineGridValueY" property.

1.3.6.12 FlashBackColor Property

TRUE, when flashing of the background is activated. BOOLEAN write-read access

1.3.6.13 FlashBorderColor Property

TRUE, when flashing of the object lines is activated. BOOLEAN write-read access.

1.3.6.14 FlashFlashPicture Property

TRUE, when flashing of the flash picture is activated. BOOLEAN write-read access.

1.3.6.15 FlashForeColor Property

TRUE, when flashing of the text is activated. BOOLEAN write-read access.

1.3.6.16 FlashPicReferenced Property

TRUE, when the assigned flash picture should be saved. Otherwise, only the associated object reference is saved. Read only access.

1.3.6.17 FlashPicTransColor Property

Defines which color of the bitmap object (.bmp, .dib) assigned to the flash picture should be set to "transparent". LONG write-read access.

The color is then only set to "transparent" when the "FlashPicUseTransColor" property has the value TRUE.

1.3.6.18 FlashPicture Property

Returns the flash picture. Read only access.

The picture (*.BMP or *.DIB) must be located in the "GraCS" directory of the current project so that it can be integrated.

In this context, the "FlashPicReferenced" property defines whether the flash picture should be saved together with the object status display or referenced.

1.3.6.19 FlashPicUseTransColor Property

TRUE, when the configured color ("FlashPicTransColor" property) of the bitmap objects assigned to the flash picture should be set to "transparent". BOOLEAN write-read access.

1.3.6.20 FlashRate Property

Defines or returns the flash frequency. Value range from 0 to 2.

0 = slow

1 = medium

2 = fast

1.3.6.21 FlashRateBackColor Property

Defines or returns the flash frequency for the object background. Value range from 0 to 2.

0 = slow

1 = medium

2 = fast

1.3.6.22 FlashRateBorderColor Property

Defines or returns the flash frequency for the lines of the object. Value range from 0 to 2.

0 = slow

1 = medium

2 = fast

1.3.6.23 FlashRateFlashPic Property

Defines or returns the flash frequency for the flash picture. Value range from 0 to 2.

0 = slow

1 = medium

2 = fast

1.3.6.24 FlashRateForeColor Property

Defines or returns the flash frequency for the object label. Value range from 0 to 2.

0 = slow

1 = medium

2 = fast

1.3.6.25 Flip Property

Mirrors the icon on the vertical and/or horizontal middle axis of the icon.

- Zero - 0: The icon is not mirrored.
- Horizontal - 1: The icon is mirrored on the vertical, middle axis.
- Vertical - 2: The icon is mirrored on the horizontal, middle axis.
- Both - 3: The icon is mirrored both on the horizontal and vertical middle axes.

1.3.6.26 FocusColor Property

If the focus is positioned on the control in runtime, the labeling and position text are identified by a border. FocusColor defines the color of the border.

1.3.6.27 FocusRect Property

TRUE, when the button should be provided with a selection border, in runtime, as soon as it receives the focus. BOOLEAN write-read access.

1.3.6.28 FocusWidth Property

If the focus is positioned on the control in runtime, the labeling and position text are identified by a border. FocusWidth defines the width of the border, value range of 1-10 pixels. LONG write-read access.

1.3.6.29 Font Property

Defines or returns the font. Write/Read access.

The font object has the following sub-properties

- Size
- Bold (yes/no)
- Name (font name)
- Italic (yes/no)
- Underline (yes/no)
- StrikeThrough (yes/no)

If two font properties are directly assigned, only the default property "Name" is assumed.

Example

```
'VBS74
Dim objControl1
Dim objControl2
Set objControl1 = ScreenItems("Control1")
Set objControl2 = ScreenItems("Control2")
objControl2.Font = objControl1.Font ' take over only the type of font
```

1.3.6.30 FontBold Property

TRUE, when the text in the object should be assigned the "bold" attribute. BOOLEAN write-read access.

1.3.6.31 FontItalic Property

TRUE, when the text in the object should be assigned the "italic" attribute.
BOOLEAN write-read access.

1.3.6.32 FontName Property

Defines or returns the font name of the text in the object.
All the fonts installed in Windows are available for selection.

1.3.6.33 FontPosition Property

Returns the font name for the display of the slider position in the bottom part of the object. All the fonts installed in Windows are available for selection. Read only access.

1.3.6.34 FontSize Property

Defines or returns the font size of the text in the object in points.

1.3.6.35 FontStrikeThru Property

TRUE, when the text in the object should be assigned the "strikethrough" attribute.
BOOLEAN write-read access.

1.3.6.36 FontUnderline Property

TRUE, when the text in the object should be assigned the "underline" attribute.
BOOLEAN write-read access.

1.3.6.37 ForeColor Property

Defines or returns the color of the font for the text in the object. LONG write-read access.

1.3.6.38 ForeFlashColorOff Property

Defines or returns the color of the text for flash status "Off". LONG write-read access.

1.3.6.39 ForeFlashColorOn Property

Defines or returns the color of the text for flash status "On". LONG write-read access.

1.3.6.40 FrameColor Property

Defines or returns the color of the rectangular or square area located on the graduated scale disk. LONG write-read access.

1.3.6.41 FrameColorDown Property

Defines or returns the color for the right, bottom part of the 3D frame of the button (button pressed). LONG write-read access.

1.3.6.42 FrameColorUp Property

Defines or returns the color for the left, top part of the 3D frame of the button (button not pressed). LONG write-read access.

1.3.6.43 FramePicture Property

Returns the picture name of the background picture for the graduated scale disk. Read only access.

1.3.6.44 FrameScale Property

Defines or returns the diameter of the graduated scale disk in relation to smallest value of the width and height geometric attributes. Write/Read access.

The value range is (scale distance - scale width) to 1.

1.3.6.45 **FrameWidth Property**

Defines or returns the frame width of the button in pixels. Write/Read access.

1.3.6.46 **FreezeProviderConnections Property**

Enables modification of the data connection properties ("ProviderType", "Source"...), without the changes taking immediate effect. On changing "SourceTagNameX", for example, impermissible combinations can be created with "SourceTagNameY".

Therefore, "FreezeProviderConnections" must be set to TRUE before modifying a data connection attribute. After modifying all the data connection, "FreezeProviderConnection" is set to FALSE and the changes take effect.

1.3.7 **G**

1.3.7.1 **OraphDirection Property**

Defines which edge of the trend window should display the current values. Write/Read access.

0: Positive values run to the right and upwards.

-1: Positive values run to the left and upwards.

-2: Positive values run to the right and upwards.

-3: Positive values run to the right and downwards.

1.3.7.2 **GridLineHorz Property**

TRUE, when the message window columns are separated by horizontal separating lines. BOOLEAN write-read access.

1.3.7.3 **GridLines Property**

TRUE, when the trend window is displayed with grid lines parallel to the X-axis. The distance between two grid lines can be changed using the "GridLineValue" property. BOOLEAN write-read access.

1.3.7.4 GridLineValue Property

Defines the distance between two grid lines. Whether the information is evaluated is dependent on the value of the "GridLines" property.

1.3.7.5 GridlinesValueX Property

Defines or returns the distance between two grid lines on the X-axis. Whether the information is evaluated is dependent on the value of the "GridLinesX" property.

1.3.7.6 GridlinesValueY Property

Defines or returns the distance between two grid lines on the Y-axis. Whether the information is evaluated is dependent on the value of the "GridLinesY" property.

1.3.7.7 GridLineVert Property

TRUE, when the message window columns are separated by vertical separating lines. BOOLEAN write-read access.

1.3.7.8 GridlinesX Property

TRUE, when the trend window is displayed with grid lines parallel to the X-axis. The distance between two grid lines can be changed using the "GridLineValueX" property.

1.3.7.9 GridlinesY Property

TRUE, when the trend window is displayed with grid lines parallel to the Y-axis. The distance between two grid lines can be changed using the "GridLineValueX" property.

1.3.8 H

1.3.8.1 andFillColor Property

Defines or returns the fill color of all the hands in the analog clock. In order that the hands are displayed with the fill color defined, the "Handtype" property must be set to "0" (covering). LONG write-read access.

1.3.8.2 Handtype Property

Defines the representation of the hands:

- 0: The hands are filled in the hand color defined and the edges in the foreground color.
- 1: The hands fill color is transparent and the edges displayed in the foreground color.

1.3.8.3 Height Property

Defines or returns the height of the object in pixels.

LONG (write-read access)

Example

The following example halves the height of all objects in the "NewPDL1" picture whose names begin with "Circle":

```
'VBS75
Dim objScreen
Dim objCircle
Dim lngIndex
Dim strName
lngIndex = 1
Set objScreen = HMIRuntime.Screens("NewPDL1")
For lngIndex = 1 To objScreen.ScreenItems.Count
    '
    'Searching all circles
    strName = objScreen.ScreenItems.Item(lngIndex).ObjectName
    If "Circle" = Left(strName, 6) Then
        '
        'to halve the height of the circles
        Set objCircle = objScreen.ScreenItems(strName)
        objCircle.Height = objCircle.Height / 2
    End If
Next
```

1.3.8.4 HiddenInput Property

TRUE, when the input value should not be displayed when being entered. Each character entered is substituted by a *. BOOLEAN write-read access.

1.3.8.5 Hotkey Property

Returns the function key related to the mouse operation in respect of a button object.

Read only access.

1.3.8.6 HourNeedleHeight Property

Defines or returns the length of the hour hand for the analog clock. The specification of the length is entered as a percentage value in relation to half the length of the short side of the rectangular background. Write/Read access.

Example:

The short side of the rectangular background is 100 pixels long.

The hour hand length is specified as 50.

Therefore, the length of the hour hand is $(100 \text{ pixel} / 2) * 0.5 = 25 \text{ Pixel}$.

1.3.8.7 HourNeedleWidth Property

Defines or returns the width of the hour hand for the analog clock. The width is specified as a percentage value related to double the length of the hour hand. Write/Read access.

Example:

The length of the hour hand is 25 pixels.

The hour hand width is specified as 10.

Therefore, the width of the hour hand is $25 \text{ pixel} * 2 * 0.1 = 5 \text{ Pixel}$.

1.3.8.8 Hysteresis Property

TRUE, when the display should appear with hysteresis. BOOLEAN write-read access.

1.3.8.9 HysteresisRange Property

Defines or returns the hysteresis in a %age of the display value.

The Hysteresis property must be set to TRUE for the hysteresis to be calculated.

1.3.9 Index Property

Check Box, Radio Box

Defines or returns the number (0 to 31) of the field whose text is to be defined.

Polygon, Polyline

Defines or returns the number of the corner point whose position coordinates are to be modified or displayed.

WinCC Online Trend Control, WinCC Online Table Control, WinCC Function Trend Control

The "Index" property is evaluated by other properties in order to be able to assign the settings to a specific trend or column pair. The valid values for the index move within the range from 0 to (NumItems - 1). The "NumItems" properties contains the number of the trends/column pairs to be displayed.

Status Display

Defines or returns the status (0 to 255). A basic picture and flash picture can be defined for each status value.

1.3.9.1 **InnerBevelOffset Property**

Defines the distance between the inner and outer bevels.

1.3.9.2 **InnerBevelStyle Property**

Defines the 3D effect for the inner bevel of the object.

- 0: No border.
- 1: The border is displayed depressed.
- 2: The border is displayed raised.
- 3: The border is displayed in one color without a 3D effect. The border color is defined by the "BevelColorDown" property.

1.3.9.3 **InnerBevelWidth Property**

Defines the width of the inner bevel in pixels.

1.3.9.4 InsertData Property

Inserts data for the current trend.

TRUE: "DataIndex" is ignored and the data is appended to that in the data buffer.

FALSE: The data is inserted at the "DataIndex" position in the data buffer.

The trend window is redrawn following each operation involving "Insert Data".

1.3.9.5 ItemBorderBackColor Property

Defines or returns the background color for separation lines in the selection list of the text list object. LONG write-read access. The background color is only visible with the property setting ItemBorderStyle > 0.

1.3.9.6 ItemBorderColor Property

Defines or returns the color for separation lines in the selection list of the text list object. LONG write-read access.

1.3.9.7 ItemBorderStyle Property

Defines or returns the color for the separation line style in the selection list of the text list object. Value range from 0 to 4.

0 = solid line

1 = dashed line

2 = dotted line

3 = dash-dotted line

4 = dash-dot-dot line

1.3.9.8 ItemBorderWidth Property

Defines or returns the dividing line weight in pixels in the selection list of the text list object.

1.3.9.9 ItemVisible Property

TRUE, when a trend or a column pair reference by the "Index" property is visible.
BOOLEAN write-read access.

1.3.10 L

1.3.10.1 **abel Property**

The "Index" property references a trend. Label is used to define the name of the time axis or value axis in accordance with the value of the "TimeAxis" property.

1.3.10.2 **LabelColor Property**

Defines the color of the scale label.

1.3.10.3 **LabelX Property**

Defines or returns the label on the X-axis for a trend referenced by "Index" according to the value of "TimeAxisX". Write/Read access.

1.3.10.4 **LabelY Property**

Defines or returns the label on the Y-axis for a trend referenced by "Index" according to the value of "TimeAxisY". Write/Read access.

1.3.10.5 **LanguageSwitch Property**

Returns the value which defines where the language dependent assigned texts are stored. Read only access.

TRUE, when the texts in the Text Library are managed. Translation to other language occurs in the Text Library.

FALSE, when the texts are managed directly in the object. Translation to other language can be carried out using EasyLanguage.

1.3.10.6 **Language Property**

Defines the current runtime language or reads it.

The runtime language is specified in VBS by means of a language code, e.g. 1031 for German - Germany, 1033 for English - United States etc. An overview of all language codes is provided in the principles of VBScript under the topic "Locale ID (LCID) Chart".

INTEGER (write-read access)

Example

The following example sets the data language to German:

```
'VBS76  
HMIRuntime.Language = 1031
```

1.3.10.7 **LastError Property**

Returns an error code regarding the success of the last operation, e.g. information on a tag write or read process. The "QualityCode" property can provide information

on the quality of the returned value. A of the error can be called in using the "Error" property.

LONG (read only)

The following error codes are defined:

OK	0x00000000
OPERATION_ERROR	0x80040001
VARIABLE_ERROR	0x80040002

In order that LastError returns a value, a read must be executed beforehand.

Example

The following example displays the error code for "Tag1":

```
'VBS77
Dim objTag
Set objTag = HMIRuntime.Tags("Tag1")
objTag.Read
MsgBox objTag.LastError
```

1.3.10.8 Layer Property

Returns the layer of the picture in which the object is located. There is a total of 32 layers available, whereby Layer "0" is the bottom layer and Layer "31" the top layer.

The configured objects are initially in the background of a layer.

LONG (read only)

Note:

The Layer Property defines the layer in which the object is located. Layer 0 is displayed as layer "0". During access using VBS, the layers are counted up from 1. Therefore, layer 1 must be referred to as layers(2).

Example

The following example displays the name and layer of all the objects in the picture "NewPDL1":

```
'VBS78
Dim objScreen
Dim objScrltem
Dim lngAnswer
Dim lngIndex
Dim strName
lngIndex = 1
Set objScreen = HMIRuntime.Screens("NewPDL1")
For lngIndex = 1 To objScreen.ScreenItems.Count
strName = objScreen.ScreenItems.Item(lngIndex).ObjectName
Set objScrltem = objScreen.ScreenItems(strName)
lngAnswer = MsgBox(strName & " is in layer " & objScrltem.Layer,vbOKCancel)
If vbCancel = lngAnswer Then Exit For
Next
```

1.3.10.9 Layer00Checked Property

TRUE, when the 0 limit should be monitored. BOOLEAN write-read access. Limit value and representation are defined with the Layer00Value and Layer00Color properties.

1.3.10.10 Layer01Checked Property

TRUE, when the 1 limit should be monitored. BOOLEAN write-read access. Limit value and representation are defined with the Layer01Value and Layer01Color properties.

1.3.10.11 Layer02Checked Property

TRUE, when the 2 limit should be monitored. BOOLEAN write-read access. Limit value and representation are defined with the Layer02Value and Layer02Color properties.

1.3.10.12 Layer03Checked Property

TRUE, when the 3 limit should be monitored. BOOLEAN write-read access. Limit value and representation are defined with the Layer03Value and Layer03Color properties.

1.3.10.13 Layer04Checked Property

TRUE, when the 4 limit should be monitored. BOOLEAN write-read access.
Limit value and representation are defined with the Layer04Value and Layer04Color properties.

1.3.10.14 Layer05Checked Property

TRUE, when the 5 limit should be monitored. BOOLEAN write-read access.
Limit value and representation are defined with the Layer05Value and Layer05Color properties.

1.3.10.15 Layer06Checked Property

TRUE, when the 6 limit should be monitored. BOOLEAN write-read access.
Limit value and representation are defined with the Layer06Value and Layer07Color properties.

1.3.10.16 Layer07Checked Property

TRUE, when the 7 limit should be monitored. BOOLEAN write-read access.
Limit value and representation are defined with the Layer07Value and Layer07Color properties.

1.3.10.17 Layer08Checked Property

TRUE, when the 8 limit should be monitored. BOOLEAN write-read access.
Limit value and representation are defined with the Layer08Value and Layer08Color properties.

1.3.10.18 Layer09Checked Property

TRUE, when the 9 limit should be monitored. BOOLEAN write-read access.
Limit value and representation are defined with the Layer09Value and Layer09Color properties.

1.3.10.19 Layer10Checked Property

TRUE, when the 10 limit should be monitored. BOOLEAN write-read access.
Limit value and representation are defined with the Layer10Value and Layer10Color properties.

1.3.10.20 Layer00Color Property

Defines or returns the color for limit 0. LONG write-read access.
When monitoring of the limit value is activated (Layer00Checked property), the bar turns to the color defined by this attribute on reaching the limit value.

1.3.10.21 Layer01Color Property

Defines or returns the color for limit 1. LONG write-read access.
When monitoring of the limit value is activated (Layer01Checked property), the bar turns to the color defined by this attribute on reaching the limit value.

1.3.10.22 Layer02Color Property

Defines or returns the color for limit 2. LONG write-read access.
When monitoring of the limit value is activated (Layer02Checked property), the bar turns to the color defined by this attribute on reaching the limit value.

1.3.10.23 Layer03Color Property

Defines or returns the color for limit 3. LONG write-read access.
When monitoring of the limit value is activated (Layer03Checked property), the bar turns to the color defined by this attribute on reaching the limit value.

1.3.10.24 Layer04Color Property

Defines or returns the color for limit 4. LONG write-read access.

When monitoring of the limit value is activated (Layer04Checked property), the bar turns to the color defined by this attribute on reaching the limit value.

1.3.10.25 Layer05Color Property

Defines or returns the color for limit 5. LONG write-read access.

When monitoring of the limit value is activated (Layer05Checked property), the bar turns to the color defined by this attribute on reaching the limit value.

1.3.10.26 Layer06Color Property

Defines or returns the color for limit 6. LONG write-read access.

When monitoring of the limit value is activated (Layer06Checked property), the bar turns to the color defined by this attribute on reaching the limit value.

1.3.10.27 Layer07Color Property

Defines or returns the color for limit 7. LONG write-read access.

When monitoring of the limit value is activated (Layer07Checked property), the bar turns to the color defined by this attribute on reaching the limit value.

1.3.10.28 Layer08Color Property

Defines or returns the color for limit 8. LONG write-read access.

When monitoring of the limit value is activated (Layer08Checked property), the bar turns to the color defined by this attribute on reaching the limit value.

1.3.10.29 Layer09Color Property

Defines or returns the color for limit 9. LONG write-read access.

When monitoring of the limit value is activated (Layer09Checked property), the bar turns to the color defined by this attribute on reaching the limit value.

1.3.10.30 Layer10Color Property

Defines or returns the color for limit 10. LONG write-read access.

When monitoring of the limit value is activated (Layer10Checked property), the bar turns to the color defined by this attribute on reaching the limit value.

1.3.10.31 Layer00Value Property

Defines or returns the value for "Limit 0".

Monitoring only takes effect when the Layer00Checked property value is set to TRUE.

1.3.10.32 Layer01Value Property

Defines or returns the value for "Limit 1".

Monitoring only takes effect when the Layer01Checked property value is set to TRUE.

1.3.10.33 Layer02Value Property

Defines or returns the value for "Limit 2".

Monitoring only takes effect when the Layer02Checked property value is set to TRUE.

1.3.10.34 Layer03Value Property

Defines or returns the value for "Limit 3".

Monitoring only takes effect when the Layer03Checked property value is set to TRUE.

1.3.10.35 Layer04Value Property

Defines or returns the value for "Limit 4".

Monitoring only takes effect when the Layer04Checked property value is set to TRUE.

1.3.10.36 Layer05Value Property

Defines or returns the value for "Limit 5".

Monitoring only takes effect when the Layer05Checked property value is set to TRUE.

1.3.10.37 Layer06Value Property

Defines or returns the value for "Limit 6".

Monitoring only takes effect when the Layer06Checked property value is set to TRUE.

1.3.10.38 Layer07Value Property

Defines or returns the value for "Limit 7".

Monitoring only takes effect when the Layer07Checked property value is set to TRUE.

1.3.10.39 Layer08Value Property

Defines or returns the value for "Limit 8".

Monitoring only takes effect when the Layer08Checked property value is set to TRUE.

1.3.10.40 Layer09Value Property

Defines or returns the value for "Limit 9".

Monitoring only takes effect when the Layer09Checked property value is set to TRUE.

1.3.10.41 Layer10Value Property

Defines or returns the value for "Limit 10".

Monitoring only takes effect when the Layer10Checked property value is set to TRUE.

1.3.10.42 Left Property

Defines or returns the X-coordinate of an object (measured from the top left edge of the picture) in pixels. The X-coordinate relates to the top left corner of the rectangle enclosing the object.

LONG (write-read access)

Example

The following example shifts all objects in the picture "NewPDL1" 5 pixels to the left:

```
'VBS79
Dim objScreen
Dim objScrltem
Dim lngIndex
Dim strName
lngIndex = 1
Set objScreen = HMIRuntime.Screens("NewPDL1")
For lngIndex = 1 To objScreen.ScreenItems.Count
  strName = objScreen.ScreenItems.Item(lngIndex).ObjectName
  Set objScrltem = objScreen.ScreenItems(strName)
  objScrltem.Left = objScrltem.Left - 5
Next
```

1.3.10.43 LeftComma Property

Defines or returns the number of digits to the left of the decimal point (0 to 20).

1.3.10.44 LightEffect Property

TRUE, when the light effect should be activated. BOOLEAN write-read access.

1.3.10.45 LimitHigh4 Property

Defines or returns the upper limit value for "Reserve 4".

The CheckLimitHigh4 property must be set to TRUE in order that the "Reserve 4" limit value can be monitored.

The type of the evaluation (in percent or absolute) is defined in the TypeLimitHigh4 property.

1.3.10.46 LimitHigh5 Property

Defines or returns the upper limit value for "Reserve 5".

The CheckLimitHigh5 property must be set to TRUE in order that the "Reserve 5" limit value can be monitored.

The type of the evaluation (in percent or absolute) is defined in the TypeLimitHigh5 property.

1.3.10.47 LimitLow4 Property

Defines or returns the lower limit value for "Reserve 4".

The CheckLimitLow4 property must be set to TRUE in order that the "Reserve 4" limit value can be monitored.

The type of the evaluation (in percent or absolute) is defined in the TypeLimitLow4 property.

1.3.10.48 LimitLow5 Property

Defines or returns the lower limit value for "Reserve 5".

The CheckLimitLow5 property must be set to TRUE in order that the "Reserve 5" limit value can be monitored.

The type of the evaluation (in percent or absolute) is defined in the TypeLimitLow5 property.

1.3.10.49 LimitMax Property

Defines or returns the upper limit value as an absolute value depending on the data format.

If the value to be displayed exceeds the upper limit value, it is identified by a series of *** , indicating it cannot be displayed.

1.3.10.50 LimitMin Property

Defines or returns the lower limit value as an absolute value depending on the data format.

If the value to be displayed exceeds the upper limit value, it is identified by a series of *** , indicating it cannot be displayed.

1.3.10.51 LineFont Property

TRUE, when the font size should be automatically adapted to the line height. BOOLEAN write-read access.

1.3.10.52 LineHeight Property

TRUE, when the line height can be modified. BOOLEAN write-read access.

1.3.10.53 LineTitle Property

TRUE, when the message window a column with consecutive number contains queued messages. BOOLEAN write-read access.

1.3.10.54 ListType Property

Returns the data type displayed in the case of a text list object. Read only access.

Value range from 0 to 2.

0 = decimal

1 = binary

2 = bit

1.3.10.55 LoadDataImmediately Property

TRUE, when the tag values for the time range to be displayed are loaded from the archives on opening a picture. BOOLEAN write-read access.

1.3.10.56 LocaleID Property

Defines the language to be displayed in the control, e.g. 1031 for German. Write/Read access.

The full list of language codes is available in the WinCC documentation (Index > Language ID).

1.3.10.57 LockBackColor Property

Defines or returns the background color of the button for a locked measuring point. LONG write-read access.

The LockStatus property must be set to TRUE for the background color to be displayed.

1.3.10.58 LockStatus Property

TRUE, when a locked measuring point should be displayed. BOOLEAN write-read access.

1.3.10.59 LockText Property

Defines the label of the button for a locked measuring point.
The LockStatus property must be set to TRUE for the label to be displayed.

1.3.10.60 LockTextColor Property

Defines or returns the color of the button label for a locked measuring point. LONG write-read access.
The LockStatus property must be set to TRUE for the background color to be displayed.

1.3.10.61 LongStrokesBold Property

TRUE, when the long sections of a scale should be displayed in bold face. BOOLEAN write-read access.

1.3.10.62 LongStrokesOnly Property

TRUE, when only the long sections of a scale should be displayed . BOOLEAN write-read access.

1.3.10.63 LongStrokesSize Property

Defines or returns the length of the axis section in pixels.

1.3.10.64 LongStrokesTextEach Property

Returns the value which defines which sections of the scale displayed should be labeled (1 = every section, 2 = every second section, etc.). Read only access

1.3.10.65 LowerLimit Property

WinCC Online Trend Control/WinCC Function Trend Control

TRUE, when the "LowerLimitColor" specification is to be used in order to identify the tag values (from a trend referenced via "Index") which lie below the value defined in "LowerLimitValue". BOOLEAN write-read access.

WinCC Online Table Control

The value of this attribute cannot be changed. Read only access.

1.3.10.66 LowerLimitColor Property

WinCC Online Trend Control/WinCC Function Trend Control

Defines the color to be used in order to identify the tag values (from trend referenced via "Index") which lie below the value defined in "LowerLimitValue". Whether the information is evaluated is dependent on the value of the "LowerLimit" property. The color is defines as an RGB value. LONG write-read access.

Online Table Control

The value of this attribute cannot be changed. Read only access.

1.3.10.67 LowerLimitValue Property

WinCC Online Trend Control/WinCC Function Trend Control

Tag values (from a trend referenced via "Index") which lie below the value defined by "LowerLimitValue" are identified by the color specified in "LowerLimitColor". Whether the information is evaluated is dependent on the value of the "LowerLimit" attribute.

Online Table Control

The value of this attribute cannot be changed. Read only access.

1.3.11 M

1.3.11.1 Marker Property

TRUE, when the limit values should be displayed as scale values. BOOLEAN write-read access.

1.3.11.2 Max Property

Defines or returns the absolute value in the case of a full value display. This value is displayed if the scale display is active.

1.3.11.3 MaximizeButton Property

TRUE, when the object can be maximized in runtime. Read only access.

1.3.11.4 MCGUBackColorOff Property

Defines or returns the background color for flash status "Off" in the case of the "Departed Unacknowledged" status. LONG write-read access.

1.3.11.5 MCGUBackColorOn Property

Defines or returns the background color for flash status "On" in the case of the "Departed Unacknowledged" status. LONG write-read access.

1.3.11.6 MCGUBackFlash Property

TRUE, when the background should flash when a message departs unacknowledged. BOOLEAN write-read access.

1.3.11.7 MCGUTextColorOff Property

Defines or returns the color of the text for flash status "Off" in the case of the "Departed Unacknowledged" status. LONG write-read access.

1.3.11.8 MCGUTextColorOn Property

Defines or returns the background color of the text for flash status "Off" in the case of the "Departed Unacknowledged" status. LONG write-read access.

1.3.11.9 MCGUTextFlash Property

TRUE, when the font should flash when a message departs unacknowledged. BOOLEAN write-read access.

1.3.11.10 MCKOBackColorOff Property

Defines or returns the background color for flash status "Off" in the case of the "Arrived" status. LONG write-read access.

1.3.11.11 MCKOBackColorOn Property

Defines or returns the background color for flash status "On" in the case of the "Arrived" status. LONG write-read access.

1.3.11.12 MCKOBackFlash Property

TRUE, when the background should flash when a message arrives. BOOLEAN write-read access.

1.3.11.13 MCKOTextColorOff Property

Defines or returns the color of the text for flash status "Off" in the case of the "Arrived" status. LONG write-read access.

1.3.11.14 MCKOTextColorOn Property

Defines or returns the background color of the text for flash status "On" in the case of the "Arrived" status. LONG write-read access.

1.3.11.15 MCKOTextFlash Property

TRUE, when the font should flash when a message arrives. BOOLEAN write-read access.

1.3.11.16 MCKQBackColorOff Property

Defines or returns the background color for flash status "Off" in the case of the "Departed Acknowledged" status. LONG write-read access.

1.3.11.17 MCKQBackColorOn Property

Defines or returns the background color for flash status "On" in the case of the "Departed Acknowledged" status. LONG write-read access.

1.3.11.18 MCKQBackFlash Property

TRUE, when the background should flash when a message departs acknowledged. BOOLEAN write-read access.

1.3.11.19 MCKQTextColorOff Property

Defines or returns the color of the text for flash status "Off" in the case of the "Departed Acknowledged" status. LONG write-read access.

1.3.11.20 MCKQTextColorOn Property

Defines or returns the background color of the text for flash status "On" in the case of the "Departed Acknowledged" status. LONG write-read access.

1.3.11.21 MCKQTextFlash Property

TRUE, when the font should flash when a message departs acknowledged. BOOLEAN write-read access.

1.3.11.22 MCText Property

Defines or returns the label for the respective message class.

1.3.11.23 MeasurePoints Property

The "Index" property references a trend. "MeasurePoints" defines the number of measuring points to be displayed. The information is only evaluated when the "TimeAxis" property is set to the value "-1".

1.3.11.24 MessageClass Property

Defines the respective message type (Alarm High, Alarm Low, Warning High, Warning Low, ...) for which the "Display Text", "Arrived-", "Arrived Acknowledged -" and "Departed Unacknowledged -" settings have been configured.

1.3.11.25 Min Property

Defines or returns the absolute value in the case of the smallest value display. This value is displayed if the scale display is active.

1.3.11.26 MinuteNeedleHeight Property

Defines or returns the length of the minute hand for the analog clock. The specification of the length is entered as a percentage value in relation to half the length of the short side of the rectangular background. Write/Read access.

Example:

The short side of the rectangular background is 100 pixels long.

The minute hand width is specified as 80.

Therefore, the length of the minute hand is $(100 \text{ pixel} / 2) * 0.8 = 40 \text{ pixel}$.

1.3.11.27 MinuteNeedleWidth Property

Defines or returns the width of the minute hand for the analog clock. The width is specified as a percentage value related to double the length of the minute hand.

Example:

The length of the minute hand is 40 pixels.

The minute hand width is specified as 8.

Therefore, the width of the minute hand is $40 \text{ pixel} * 2 * 0.08 = 6 \text{ pixels (rounded)}$.

1.3.11.28 Moveable Property

TRUE, when the object can be moved in runtime. Read only access.

1.3.11.29 MsgCtrlFlags Property

Defines the sorting sequence in Alarm Control. Write/Read access.

- 0: Sorting is defined according to the entries in the time column in ascending order, i.e. the oldest messages are at the top of the message window.
- 1: Sorting is defined according to the entries in the time column in descending order, i.e. the oldest messages are at the bottom of the message window, the latest at the top. In the case of this value, the "AutoScroll" property is automatically deactivated, otherwise the current message could be moved out of the display area of the message window.

1.3.11.30 MsgFilterSQL Property

Defines an SQL Statement to the selected messages displayed in the message window. Write/Read access.

1.3.12 N

1.3.13 Name Property

of layer and tag object

Returns the object name. STRING (read only)

- In the case of tags, the name of the tag without server and tag prefix
- In the case of layers, the layer name

Tags

The tag "Name" property is used to address the tag via the tag list. The name of a tag can contain a server prefix. In WinCC, tag names are structured according to the following scheme:

<Server prefix>::<Tag prefix><Tag Name>

If the tag name alone is specified, the server prefix and tag prefix are removed from the context of the picture.

If the tag is specified with a server prefix in the tag name, the tags and server prefix of the context are ignored and the server prefix included is used.

WinCC Function Trend Control

The "Index" property references a trend. "Name" defines the name of the trend.

1.3.13.1 NeedleColor Property

Defines or returns the color of the pointer. LONG write-read access.

1.3.13.2 NormalColor Property

Defines the color of the normal area of the scale. LONG write-read access.

1.3.13.3 NumberLines Property

Defines or return the number of lines the text list object should contain. If the amount of configured text is larger than this value, the selection list receives a vertical scroll bar.

1.3.13.4 NumItems Property

Returns the number of trends or column pairs (visible and invisible) in the window which have been configured. Write/Read access.

1.3.14 O

1.3.14.1 Object Property

If a non-WinCC control is used, it is possible that the properties provided by the control have the same names as the general ScreenItem properties. In such cases, the ScreenItem properties have priority. The "hidden" properties of an external control supplier can be accessed using the additional "object" property.

Example

Address the properties of an external control supplier as follows:

Control.object.type

If the following form alone is used

Control.type

the properties of the ScreenItem object are used in the case of identical names.

1.3.14.2 ObjectName Property

Returns the object name.

- In the case of graphic objects, the object name
- In the case of pictures, the picture name

STRING (read only)

Example

The following example issues the names of all the objects contained in the picture "NewPDL1":

```
'VBS80
Dim objScreen
Dim lngIndex
Dim lngAnswer
Dim strName
lngIndex = 1
Set objScreen = HMIRuntime.Screens("NewPDL1")
For lngIndex = 1 To objScreen.ScreenItems.Count
strName = objScreen.ScreenItems.Item(lngIndex).ObjectName
lngAnswer = MsgBox("Name of object " & lngIndex & ": " & strName, vbOKCancel)
If vbCancel = lngAnswer Then Exit For
Next
```

Pictures

Establish the picture name directly from the "ObjectName" property:

```
'VBS81
MsgBox "Screenname: " & HMIRuntime.ActiveScreen.ObjectName
```

1.3.14.3 OffsetLeft Property

Defines or returns the distance of the picture from the left edge of the picture window.

1.3.14.4 OffsetTop Property

Defines or returns the distance of the picture from the top edge of the picture window.

1.3.14.5 Online Property

Serves to start or stop updating.

- 0: The updating for the display is stopped. The values are buffered and entered when the button is operated again.
- -1: The updated display is resumed.

1.3.14.6 OnTop Property

TRUE, when the object should remain in the foreground in runtime. Read only access.

1.3.14.7 OperationMessage Property

TRUE, when a message should be issued following successful operation. BOOLEAN write-read access.

The operation is sent to the diagnostics system and archived there. The message system can retrieve the operation from there in order, for example, to display the message in a message line.

Special features of I/O field, text list and slider

The reason for the operation can only be entered when the "OperationReport" property has been set to TRUE.

1.3.14.8 OperationReport Property

TRUE, when the reason for an operation should be recorded. BOOLEAN write-read access.

When an operation is subsequently initiated on the object in runtime, a dialog appears in which the user can enter the reason for the operation in text form. The operation is sent to the diagnostics system and archived there.

1.3.14.9 Orientation Property

TRUE, when the text in the object should be displayed horizontally. BOOLEAN write-read access.

Description of the "Connector" object type

Modifies the orientation of the connector. BOOLEAN write-read access.

1.3.14.10 OuterBevelStyle Property

Defines the 3D effect for the outer bevel of the object.

- 0: No border.
- 1: The border is displayed depressed.
- 2: The border is displayed raised.
- 3: The border is displayed in one color without a 3D effect. The border color is defined by the "BevelColorUp" property.

1.3.14.11 OuterBevelWidth Property

Defines the width of the outer bevel in pixels.

1.3.14.12 Outline Property

TRUE, when the button should be given a black border in addition to the 3D border. BOOLEAN write-read access.

1.3.14.13 OutputFormat Property

Returns the value for the representation of the output value. The representation is dependent on the data format. Read only access.

1.3.14.14 OutputValue Property

Defines or returns presetting for the value to be displayed.

This value is used in runtime when the associated tag cannot be connected or updated when a picture is started.

1.3.15 P-Q

1.3.15.1 Parent Property

Returns the picture in which the object is embedded.

ScreenItem (read only)

Example

The following example issues the name of the basic picture:

```
'VBS82
Dim objScrItem
Set objScrItem = HMIRuntime.Screens(1).ScreenItems(1)
MsgBox "Name of BaseScreen: " & objScrItem.Parent.ObjectName
```

1.3.15.2 PasswordLevel Property

Defines the authorization for operation (e.g. no input or no triggering actions) of the object.

1.3.15.3 PersistentRT Property

TRUE, when modified window settings should be retained following a change of picture. Whether the information is evaluated is dependent on the value of the "AllowPersistence" property.

1.3.15.4 PersistentRTCS Property

TRUE, when modified settings should be retained following a change of picture and applied in the configuration system. Whether the information is evaluated is dependent on the value of the "AllowPersistence" property. BOOLEAN write-read access.

1.3.15.5 PersistentRTCSPermission Property

Defines the operator permission which is necessary in order to modify settings related to the persistence. The value to be entered relates to the number of the required permission in the User Administrator. Whether the information is evaluated is dependent on the value of the "AllowPersistence" property.

1.3.15.6 PersistentRTPermission Property

Defines the operator permission which is necessary in order to modify settings related to the persistency in runtime. The value to be entered relates to the number of the required permission in the User Administrator. Whether the information is evaluated is dependent on the value of the "AllowPersistence" property.

1.3.15.7 PicDeactReferenced Property

TRUE, when the picture assigned for the "Disable" status should be saved in the RoundButton object. Otherwise, only the associated object reference is saved.
Read only access.

1.3.15.8 PicDeactTransparent Property

Defines or returns which color of the bitmap object (.bmp, .dib) assigned to the "Disabled" status should be set to "transparent". LONG write-read access.
The color is then only set to "transparent" when the "PicDeactUseTransColor" property has the value TRUE.

1.3.15.9 PicDeactUseTransColor Property

TRUE, when the transparent color defined by the "PicDeactTransparent" property for the "Disable" status should be used. BOOLEAN write-read access.

1.3.15.10 PicDownReferenced Property

TRUE, when the picture assigned for the "On" status is to be saved. Otherwise, only the associated object reference is saved. Read only access.

1.3.15.11 PicDownTransparent Property

Defines or returns which color of the bitmap object (.bmp, .dib) assigned to the "On" status should be set to "transparent". LONG write-read access.
The color is then only set to "transparent" when the "PicDownUseTransColor" property has the value TRUE.

1.3.15.12 PicDownUseTransColor Property

TRUE, when the transparent color defined by the "PicDownTransparent" property for the "On" status should be used. BOOLEAN write-read access.

1.3.15.13 PicReferenced Property

TRUE, when the assigned picture is references the object and is not saved in it. Read only access.

1.3.15.14 PicTransColor Property

Defines or returns which color of the assigned bitmap object (.bmp, .dib) should be set to "transparent". LONG write-read access.

The color is then only set to "transparent" when the "PicUseTransColor" property has the value TRUE.

1.3.15.15 Picture Property

Returns the picture name of the background picture for the rectangular background for both the analog and digital clocks. Read only access

1.3.15.16 PictureBack Property

Returns the picture name of the picture for the object background. Read only access.

1.3.15.17 PictureDeactivated Property

Defines the picture to be displayed in the "Disable" status or returns the picture name.

The picture (*.BMP or *.DIB) must be located in the "GraCS" directory of the current project so that it can be integrated.

1.3.15.18 PictureDown Property

Defines the picture to be displayed in the "On" status or returns the picture name.

The picture (*.BMP or *.DIB) must be located in the "GraCS" directory of the current project so that it can be integrated.

1.3.15.19 PictureUp Property

Defines the picture to be displayed in the "Off" status or returns the picture name.

The picture (*.BMP or *.DIB) must be located in the "GraCS" directory of the current project so that it can be integrated.

1.3.15.20 PictureName Property

Defines the picture to be displayed in the graphic object in runtime or returns the picture name.

The picture (*.BMP or *.DIB) must be located in the "GraCS" directory of the current project so that it can be integrated.

1.3.15.21 PictureSelected Property

Returns the picture name of the picture displayed in the "On" status. "AutoSize" controls the adaptation of the size of picture and buttons. Read only access.

1.3.15.22 PictureThumb Property

Returns the picture name of the background picture for the slider. Read only access.

1.3.15.23 PictureUnselected Property

Returns the picture name of the picture displayed in the "Off" status. "AutoSize" controls the adaptation of the size of picture and buttons. Read only access.

1.3.15.24 PicUpReferenced Property

TRUE, when the picture assigned for the "Off" status should be saved in the object. Otherwise, only the associated object reference is saved. Read only access.

1.3.15.25 PicUpTransparent Property

Defines or returns which color of the bitmap object (.bmp, .dib) assigned to the "Off" status should be set to "transparent". LONG write-read access.
The color is then only set to "transparent" when the "PicUpUseTransColor" property has the value TRUE.

1.3.15.26 PicUpUseTransColor Property

TRUE, when the transparent color defined by the "PicUpTransparent" property for "Off" status should be used. BOOLEAN write-read access.

1.3.15.27 PicUseTransColor Property

TRUE, when the transparent color defined by the "PicDeactTransparent" property for the "Disable" status should be used. BOOLEAN write-read access.

1.3.15.28 Position Property

Defines the presetting for the position of the slider.

This value is used as the start value in runtime.

To operate the process value linked to this attribute, it is necessary that the process value is also linked to the "Position" event. The "Position" event is located on the "Event" tab control in the topic tree under SliderCtrl\PropertyTopics\Control Properties\Value.

1.3.15.29 Precisions Property

WinCC Online Table Control

The "Index" property references a pair of columns. "Precision" defines the number of decimal places which should be shown in this value column. A maximum of 16 decimal places can be displayed.

WinCC Online Trend Control

Defines the number of decimal places with which the scale value is specified.

1.3.15.30 PrecisionX Property

Defines or returns the number of decimal places with which the scale value for the X-axis should be specified. Write/Read access.

1.3.15.31 PrecisionY Property

Defines or returns the number of decimal places with which the scale value for the Y-axis should be specified. Write/Read access.

1.3.15.32 PredefinedAngles Property

Defines or returns the depth of the display of the 3DBarGraph object. Value range from 0 to 3.

0 = cavalier

1 = isometric

2 = axionometric

3 = freely defined

1.3.15.33 Pressed Property

TRUE, when the RoundButton object is pressed. BOOLEAN write-read access.

1.3.15.34 PrintJob Property

Defines or reads out which print layout should be used for the printed output.

1.3.15.35 Process Property

Defines or returns presetting for the value to be displayed.

This value is used in runtime when the associated tag cannot be connected or updated when a picture is started.

1.3.15.36 ProjectPath Property

Contains the path and name of the associated project. Modification of "ProjectPath" is only necessary after copying a picture to another project.

1.3.15.37 ProviderClsid Property

The "Index" property references a trend. "ProviderClsid" defines whether an archive tag or an internal or external tag should be displayed in this trend.

- {416A09D2-8B5A-11D2-8B81-006097A45D48}: The trend is connected to an archive tag.
- {A3F69593-8AB0-11D2-A440-00A0C9DBB64E}: The trend is connected to an internal or external tag.

1.3.15.38 ProviderType Property

Defines the type of values to be displayed in a trend referenced by "Index". In the case of modification of "ProviderType", impermissible combinations with other attributes for data connection could be created. Therefore, before modifying "ProviderType", the immediate acceptance of the changes must be prevented using "FreezeProviderConnections".

0: Values are supplied via the API interface.

-1: Display of online or archive tags

-2: Displaying values from a user archive

1.3.15.39 PointCount Property

Defines or returns the number of corner points. Each corner point has position coordinates and is identified via an index.

1.3.15.40 QualityCode Property

Defines a standard for the quality of a tag value after being read. The quality code is provided as a 16-bit value for automatic evaluation. After a tag has been written, the value is invalid.

SHORT (read only)

Note:

A summary of the possible Quality Codes is provided in the WinCC Information System under the heading "Communication" > "Diagnostics" or "Communication" > "Quality Codes".

Example

The following example indicates the quality of the read value when no errors have occurred during the reading process:

```
'VBS83
Dim objTag
Dim lngLastErr
Set objTag = HMIRuntime.Tags("Tag1")
objTag.Read
lngLastErr = objTag.LastError
If 0 = lngLastErr Then
MsgBox objTag.QualityCode
End If
```

1.3.16 R

1.3.16.1 Radius Property

Defines or returns the radius in pixels.

1.3.16.2 RadiusHeight Property

Defines or returns the vertical radius in pixels (0 to 999).

1.3.16.3 RadiusWidth Property

Defines or returns the horizontal radius in pixels (0 to 999).

1.3.16.4 RangeMax Property

Defines the maximum absolute value for the value display.

If the "WithLabels" property has the value -1 (yes), this value is displayed on the scale.

1.3.16.5 RangeMin Property

Defines the minimum absolute value for the value display.

If the "WithLabels" property has the value -1 (yes), this value is displayed on the scale.

1.3.16.6 Rectangular Property

Defines or returns the side ratio of the rectangular background of the gauge.
BOOLEAN write-read access.

FALSE: The size of the gauge can be adjusted to any side ratio by dragging the marking points with the mouse.

TRUE: The size of the gauge can only be adjusted by dragging the marking points with the mouse. The side ratio of the background always remains 1:1.

1.3.16.7 ReferenceRotationLeft Property

Defines or returns the X-coordinate of the reference point about which the object should be rotated in runtime.

The value of the X-coordinate is relative to the object width. Enter the value in percent starting from the left edge of the rectangle enclosing the object.

1.3.16.8 ReferenceRotationTop Property

Defines or returns the Y-coordinate of the reference point about which the object should be rotated in runtime.

The value of the Y-coordinate is relative to the object width. Enter the value in percent starting from the top edge of the rectangle enclosing the object.

1.3.16.9 RelayCurves Property

TRUE, when the trends should be displayed staggered. BOOLEAN write-read access.

1.3.16.10 Relevant Property

TRUE, when the object will be taken into account when forming the group display. BOOLEAN write-read access.

1.3.16.11 Replacement Property

The "Index" property references a trend. Values, whose start value is unknown on activating runtime or for which a substitute value is used, have an unstable status. "Replacement" defines whether such values should be identified by the color defined in "ReplacementColor". BOOLEAN write-read access.

1.3.16.12 ReplacementColor Property

The "Index" property references a trend. Values, whose start value is unknown on activating runtime or for which a substitute value is used, have an unstable status. "ReplacementColor" defines the color used to identify this value. The color is defined as an RGB value. Whether the information is evaluated is dependent on the value of the "Replacement" property.

1.3.16.13 RightComma Property

Defines or returns the number of decimal places (0 to 20).

1.3.16.14 Rotation Property

The icon can be rotated on its middle axis vertical to the character layer. The attribute specifies the angle of rotation in degrees. The angle of rotation is measured counterclockwise.

- 0: The angle of rotation is 0 degrees.
- 90: The angle of rotation is 90 degrees.
- 180: The angle of rotation is 180 degrees.
- 270: The angle of rotation is 270 degrees.

1.3.16.15 RotationAngle Property

Defines or returns the rotation angle in degrees.

In runtime, the object (starting from the configured starting position) is rotated clockwise the specified value about reference point.

1.3.16.16 RoundCornerHeight Property

Defines or returns the corner radius.

Enter the value as a percentage of half the height of the object.

1.3.16.17 RoundCornerWidth Property

Defines or returns the corner radius.

Enter the value as a percentage of half the width of the object.

1.3.16.18 RulerPrecisions Property

Defines the number of decimal places to which a measured value should be displayed when it is determined using the "Display value at this position" function.

1.3.16.19 RulerPrecisionX Property

Defines the number of decimal places used by the "Display value at this position" to display the X-coordinate of a measured value. Whether the information is evaluated is dependent on the value of the "TimeAxisX" attribute.

1.3.16.20 RulerPrecisionY Property

Defines the number of decimal places used by the "Display value at this position" to display the Y-coordinate of a measured value.

1.3.17 S

1.3.17.1 SameSize Property

TRUE, when all four buttons of a Group Display object have the same size. BOOLEAN write-read access.

1.3.17.2 ScaleColor Property

Defines or returns the color of the scale. LONG write-read access.
The "Scaling" property must be set to TRUE for the color to be displayed.

1.3.17.3 ScaleTicks Property

Defines the number of segments into which the bar will be subdivided by large tick marks of the scale:
0-100: Object can be divided into a maximum of 100 segments
= 0: The optimum number of segments is set automatically.

1.3.17.4 Scaling Property

TRUE, when a scale should also be used to represent a value. BOOLEAN write-read access.

1.3.17.5 ScalingType Property

Defines or returns the type of bar scaling. Value range from 0 to 6.

0 = linear

1 = logarithmic

2 = negative logarithmic

3 = automatic (linear)

4 = tangential

5 = square

6 = cuboid

The "Scaling" property must be set to TRUE for the color to be displayed.

1.3.17.6 ScalingTypeX Property

Defines the type of scaling of the X-axis of a trend referenced with "Index". Whether the information is evaluated is dependent on the value of the "TimeAxisX" attribute.

0: Linear

-1: Logarithmically. This setting prevents the display of negative values.

-2: Logarithmically negated. This setting prevents the display of positive values.

1.3.17.7 ScalingTypeY Property

Defines the type of scaling of the Y-axis of a trend referenced with "Index".

0: Linear

-1: Logarithmically. This setting prevents the display of negative values.

-2: Logarithmically negated. This setting prevents the display of positive values.

1.3.17.8 Screen Property

Returns the screen object to be displayed in the picture window.

1.3.17.9 ScreenName Property

Defines the picture to be displayed in the picture window in runtime or returns the picture name.

Note:

Always enter picture names without the extension "PDL" for reasons of compatibility with future versions.

1.3.17.10 Screens Property

List (HMIScreens) which enable access to all open pictures in the graphical runtime system via the picture name.

Example

The following example accesses the picture "NewPDL1":

```
'VBS84
Dim objScreen
Set objScreen = HMIRuntime.Screens("NewPDL1")
```

1.3.17.11 ScreenItems Property

List (HMIScreenItems) from which all objects in a graphic picture can be addressed and counted.

ScreenItems (read only)

Example

The following example issues the number of all the objects contained in the picture "NewPDL1":

```
'VBS85
Dim objScreen
Set objScreen = HMIRuntime.Screens("NewPDL1")
Msgbox objScreen.ScreenItems.Count
```

1.3.17.12 Scrollable Property

TRUE, when the display of the DXF picture in Runtime mode supports the scrolling function. BOOLEAN write-read access.

1.3.17.13 ScrollBars Property

TRUE, when the object is equipped with a scroll bar in runtime. Read only access.

1.3.17.14 SecondNeedleHeight Property

Defines or returns the length of the second hand for the analog clock. The specification of the length is entered as a percentage value in relation to half the length of the short side of the rectangular background. Write/Read access.

Example:

The short side of the rectangular background is 100 pixels long.

The second hand length is specified as 80.

Therefore, the length of the second hand is $(100 \text{ pixel} / 2) * 0.8 = 40 \text{ pixel}$.

1.3.17.15 SecondNeedleWidth Property

Defines or returns the width of the second hand for the analog clock. The width is specified as a percentage value related to double the length of the second hand. Write/Read access.

Example:

The length of the second hand is 40 pixels.

The second hand width is specified as 2.

Therefore, the width of the second hand is $40 \text{ pixel} * 2 * 0.02 = 2 \text{ pixels (rounded)}$.

1.3.17.16 SelBGColor Property

Defines or returns the background color of the selected entry in a text list object. LONG write-read access.

1.3.17.17 SelectionMode Property

Defines whether and how a message line can be selected.

- 0 - NoSelection: Prevents the selection of a message. An acknowledgement always relates to the oldest message queued.
- 1 - Cell: Enables the selection of fields in the message line. An acknowledgement always relates to the selected message.
- 2 - Line: Enables the selection of a message line. An acknowledgement always relates to the selected message.

1.3.17.18 SelTextColor Property

Defines or returns the color of the text of the selected entry in the text list object.
LONG write-read access.

1.3.17.19 ServerData Property

The attribute can only be modified using the "Properties of WinCC Online Trend Control" dialog. Read only access.

1.3.17.20 ServerPrefix Property

Defines the server containing the picture to be displayed in the picture window in runtime or returns the server name.

Enter the server name followed by two colons: "<Server name>:". No check is made as to whether the server actually exists.

1.3.17.21 ServerNames Property

Defines the server in a distributed system to which the data in the message window should relate. The information appears in the following form:
NameServer1;NameServer2;NameServer3. Write/Read access.

1.3.17.22 ShowBar Property

TRUE, when the bar should be displayed. BOOLEAN write-read access.

1.3.17.23 ShowDanger Property

Controls the display of the "danger zone" on the instrument scale. BOOLEAN write-read access.

TRUE: The area is identified by the color defined in "DangerColor".

FALSE: The color identification of the area is switched off.

1.3.17.24 ShowDecimalPoint Property

TRUE, when the labeling of the scale section should be with decimal numbers (decimal point and one decimal place).

FALSE, when the labeling of the scale section should be with whole numbers.

BOOLEAN write-read access.

1.3.17.25 ShowNormal Property

Controls the display of the "normal zone" on the instrument scale. BOOLEAN write-read access.

TRUE: The area is identified by the color defined for normal color.

FALSE: The color identification of the area is switched off.

1.3.17.26 ShowPeak Property

Defines the display of a slave pointer to display the maximum value. BOOLEAN write-read access.

TRUE: The slave pointer is displayed.

FALSE: The slave pointer is hidden.

1.3.17.27 ShowPosition Property

TRUE, when the slider position is to be displayed. BOOLEAN write-read access.

1.3.17.28 ShowRulerImmediately Property

TRUE, when the ruler for determining the coordinate values should be displayed when opening a picture. BOOLEAN write-read access.

1.3.17.29 ShowThumb Property

TRUE, when the slider is to be displayed. BOOLEAN write-read access.

1.3.17.30 ShowValuesExponentialX Property

TRUE, when the X-coordinate of a measured value determined via the "Display value at this position" function is displayed in exponential notation by a trend referenced via "Index". Whether the information is evaluated is dependent on the value of the "TimeAxisX" property. BOOLEAN write-read access.

1.3.17.31 ShowValuesExponentialY Property

TRUE, when the Y-coordinate of a measured value determined via the "Display value at this position" function is displayed in exponential notation by a trend referenced via "Index". BOOLEAN write-read access.

1.3.17.32 ShowWarning Property

Controls the display of the "warning zone" on the instrument scale. BOOLEAN write-read access.

TRUE: The area is identified by the color defined by the warning color attribute.

FALSE: The color identification of the area is switched off.

1.3.17.33 SignificantMask Property

Is required in runtime to display the active message class with the highest priority.

The value of the SignificantMask property represents an internal system output value does not require any specific configuration by the user. Updating is initiated in runtime by clicking on the object.

1.3.17.34 Sizeable Property

TRUE, when it should be possible to change the size of the object in runtime.
BOOLEAN write-read access.

In the case of application window and picture window: Read only access

1.3.17.35 SmallChange Property

Defines how many steps the controller can be moved with one mouse click or returns the value.

1.3.17.36 SourceBeginTime Property

In the case of online tags and archive tags ("ProviderType" = -1), it defines the starting time of the time range of a trend referenced via "Index" and to be displayed in the trend window. In the case of modification of "SourceBeginTime", impermissible combinations with other attributes for data connection could be created. Therefore, before modifying "SourceBeginTime", the immediate acceptance of the changes must be prevented using "FreezeProviderConnections".

1.3.17.37 SourceEndTime Property

In the case of online tags and archive tags ("ProviderType" = -1), it defines the stopping time of the time range of a trend referenced via "Index" and to be displayed in the trend window. In the case of modification of "SourceEndTime", impermissible combinations with other attributes for data connection could be created. Therefore, before modifying "SourceEndTime", the immediate acceptance of the changes must be prevented using "FreezeProviderConnections".

1.3.17.38 SourceNumberOfUAValues Property

For values from the user archives ("ProviderType" = -2) it defines the number of values which should be loaded from the user archive for a trend referenced via "Index". In the case of modification of "SourceNumberOfUAValues", impermissible combinations with other attributes for data connection could be created. Therefore, before modifying "SourceNumberOfUAValues", the immediate acceptance of the changes must be prevented using "FreezeProviderConnections".

1.3.17.39 SourceNumberOfValues Property

The "Index" property references a trend. In the case of online tags and archive tags ("ProviderType" = -1), "SourceNumberOfValues" defines the number of values which should be displayed in the trend window. Whether the information is evaluated is dependent on the value of the "SourceTimeRange" property.

In the case of modification of "SourceNumberOfValues", impermissible combinations with other attributes for data connection could be created. Therefore, before modifying "SourceNumberOfValues", the immediate acceptance of the changes must be prevented using "FreezeProviderConnections".

1.3.17.40 SourceTagNameX Property

The "Index" property references a trend. In the case of online tags and archive tags ("ProviderType" = -1) "SourceTagNameX" defines the tag which should be displayed along the X-axis. In the case of modification of "SourceTagNameX", impermissible combinations with other attributes for data connection could be created. Therefore, before modifying "SourceTagNameX", the immediate acceptance of the changes must be prevented using "FreezeProviderConnections".

1.3.17.41 SourceTagNameY Property

The "Index" property references a trend. In the case of online tags and archive tags ("ProviderType" = -1) "SourceTagNameY" defines the tag which should be displayed along the X-axis. In the case of modification of "SourceTagNameY", impermissible combinations with other attributes for data connection could be created. Therefore, before modifying "SourceTagNameY", the immediate acceptance of the changes must be prevented using "FreezeProviderConnections".

1.3.17.42 SourceTagProviderDataX Property

The attribute can only be modified using the "Properties of WinCC Function Trend Control" dialog.

1.3.17.43 SourceTagProviderDataY Property

The attribute can only be modified using the "Properties of WinCC Function Trend Control" dialog.

1.3.17.44 SourceTimeRange Property

The "Index" property references a trend. In the case of online tags and archive tags ("ProviderType" = -1) "SourceTimeRange" defines how the time range to be displayed in the trend window is defined. In the case of modification of "SourceTimeRange", impermissible combinations with other attributes for data connection could be created. Therefore, before modifying "SourceTimeRange", the immediate acceptance of the changes must be prevented using "FreezeProviderConnections".

0: The time range to be displayed is defined by the starting time (SourceBeginTime) and the number of value pairs (SourceNumberOfValues).

-1: The time range to be displayed is defined by the starting time (SourceBeginTime) and stopping time (SourceEndTime).

1.3.17.45 SourceUAArchive Property

The "Index" property references a trend. In the case of values from the user archives ("ProviderType" = -2), "SourceUAArchive" defines the user archive from which the values should be loaded. In the case of modification of "SourceUAArchive", impermissible combinations with other attributes for data connection could be created. Therefore, before modifying "SourceUAArchive" the immediate acceptance of the changes must be prevented using "FreezeProviderConnections".

1.3.17.46 SourceUAArchiveStartID Property

The "Index" property references a trend. In the case of values from the user archives ("ProviderType" = -2), "SourceUAArchiveStartID" defines the data record from which the values should be loaded from the user archive. In the case of modification of "SourceUAArchiveStartID", impermissible combinations with other attributes for data connection could be created. Therefore, before modifying "SourceUAArchiveStartID", the immediate acceptance of the changes must be prevented using "FreezeProviderConnections".

1.3.17.47 SourceUAColumnX Property

The "Index" property references a trend. In the case of values from the user archives ("ProviderType" = -2), "SourceUAColumnX" defines the column in the user archive from which the values for the X-axis should be loaded. In the case of modification of "SourceUAColumnX", impermissible combinations with other attributes for data connection could be created. Therefore, before modifying "SourceUAColumnX", the immediate acceptance of the changes must be prevented using "FreezeProviderConnections".

1.3.17.48 SourceUAColumnY Property

The "Index" property references a trend. In the case of values from the user archives ("ProviderType" = -2), "SourceUAColumnY" defines the column in the user archive from which the values for the Y-axis should be loaded. In the case of modification of "SourceUAColumnY", impermissible combinations with other attributes for data connection could be created. Therefore, before modifying "SourceUAColumnY", the immediate acceptance of the changes must be prevented using "FreezeProviderConnections".

1.3.17.49 SquareExtent Property

TRUE, when the size of the clock should be adjustable to any side ratio by dragging the marking points with the mouse. BOOLEAN write-read access.

1.3.17.50 StartAngle Property

Defines or returns the start of the object. The information is in counterclockwise direction in degrees, beginning at the 12:00 clock position.

1.3.17.51 Statusbar Property

TRUE, when the status bar is to be displayed. BOOLEAN write-read access.

1.3.17.52 StatusbarPanels Property

Defines the elements to be displayed in the status bar. Write/Read access.

1.3.17.53 Stretch Property

Defines whether the side ratio is retained or adjustable on changing the icon size.
BOOLEAN write-read access.

- FALSE: The side ratio is retained on changing the icon size.
- TRUE: The side ratio of the icon can be adjusted parallel to changing the icon size.

1.3.18 T

1.3.18.1 agName Property

The "Index" property references a trend. "TagName" defines the tag linked to this trend. It is specified in the form "Archive Name\Tag Name" to display tags in a process value archive or "TasgName" to display an internal or external tag which is not stored in an archive.

1.3.18.2 TagPrefix Property

Defines or returns the tag prefix which is prefixed to all tags contained in the picture window object. In this way, a picture that is embedded in a picture window retains access to its own tags while another accesses other tags.

Modification of the TagPrefix takes effect when a picture is reloaded. When a picture is changed, this occurs automatically, otherwise the picture name must be reassigned.

The tag prefix can be freely defined, but must match the name of the structure tags. It must end with a stop, for example "Structure2.".

1.3.18.3 Tags Property

Enables access to tags.

Example

The following example accesses the tag "Tag1":

```
'VBS86
Dim objTag
Set objTag = HMIRuntime.Tags("Tag1")
```

1.3.18.4 TagProviderClsid Property

The "Index" property references a trend. "TagProviderClsid" defines whether this trend should display an online tag or archived value. The data is only evaluated for online tags and archive tags ("ProviderType" = -1).

{A3F69593-8AB0-11D2-A440-00A0C9DBB64E}: Online tag.

{416A09D2-8B5A-11D2-8B81-006097A45D48}: Values are read from a process value archive or a user archive.

1.3.18.5 Template Property

Returns the template for displaying the window content of the "Application Window" object. Read only access.

The following templates are possible according to the value of the property:

Window Contents = Global Script

"GSC-Diagnostics"

The application window is supplied by Global Script applications. The results of the diagnosis system are displayed.

"GSC Runtime"

The application window is supplied by Global Script applications. The analysis results regarding in runtime are displayed.

Window Contents = Print Jobs

"All Jobs"

The application window is supplied by the protocolling system. The available reports are displayed as a list.

"All Jobs - Context Menu"

The application window is supplied by the protocolling system. The available reports are displayed as a list. A context-sensitive menu enables selection of the print options as well as printing of a report.

"Job Detail View"

The application window is supplied by the protocolling system. The available reports are displayed in a selection menu. Detailed information is displayed for the selected report.

1.3.18.6 Text Property

Defines or returns the labeling for an object.

1.3.18.7 ThumbBackColor Property

Defines the color of the slider.

1.3.18.8 TicColor Property

Defines the color of the scale tick marks. LONG write-read access.

1.3.18.9 TicFont Property

Controls the display of the scale division labeling. Read only access.

The following properties can be set:

- Font
- Font Style
- Font Size
- "Strikethrough" effect
- "Underline" effect

1.3.18.10 TicOffset Property

Defines the diameter of the imaginary circle on which the scale graduation is set. The value is related to the smaller value of the geometric properties Width and Height.

The ends of the main tick marks of the scale graduation point outwards onto this circle.

Value range from 0 to 1.

0: The scale division is in the middle of the graduated scale disk.

1: The diameter of the imaginary circle for the scale tick marks is the smaller value of the geometric properties Width and Height.

1.3.18.11 TicTextColor Property

Defines the color of the labeling of the scale tick marks.

1.3.18.12 TicTextOffset Property

Defines the diameter of the imaginary circle on which the labeling of the scale tick marks is set. The value is related to the smaller value of the geometric properties Width and Height.

Value range from 0 to 1.

0: The label is in the middle of the graduated scale disk.

1: The diameter of the imaginary circle for the label is the smaller value of the geometric properties Width and Height. As a result, part of the label can lie outside the object limits and is, thus, invisible.

1.3.18.13 TicWidth Property

Defines the length of the long tick marks for the scaling. The value is related to the half the smaller value of the geometric properties Width and Height.

The length of the tick marks for fine scaling is $0.5 \times \text{scale width}$.

Value range from 0 to end of scale.

0: No scale graduation is available. The division of the scale into ranges is not visible.

Scaling distance: The scaling division ranges from the middle point of the graduated scale disk to the value defined by the scaling distance.

1.3.18.14 Ticks Property

TRUE, when the numbered face is displayed. BOOLEAN write-read access.

1.3.18.15 TicksColor Property

Defines or returns the color of the hour markings on the face of the analog clock. LONG write-read access.

1.3.18.16 TickStyle Property

This attribute defines the appearance of the scale. Value range: 0 to 3.

As a result of the automatic scaling, it is possible that, occasionally, two scale tick marks lie directly beside each other (apparently wide tick mark). This effect can be corrected by minimally lengthening or shortening the slider object.

It is also possible to completely suppress display of the scaling ("WithAxes").

1.3.18.17 TimeAxis Property

Defines whether a common time axis should be used for all trends in the trend window.

1.3.18.18 TimeAxisFormat Property

Defines the format of the information along the time axis.

- 0: The information is provided in hh:mm
- -1: The information is provided in hh:mm:ss
- -2: The information is provided in hh:mm:ss.ms
- -3: The information is provided in hh:mm (full hours)
- -4: The information is provided in hh:mm:ss (full minutes)
- -5: The information is provided in hh:mm:ss.ms (full seconds)

1.3.18.19 TimeAxisX Property

TRUE, when a common axis should be used for all trends in the trend window.
BOOLEAN write-read access.

1.3.18.20 TimeColumnAlignment Property

The "Index" property references a pair of columns. "TimeColumnAlignment" defines the alignment of the time column for this column pair.

- 0: Time values are entered aligned left.
- 1: Time values are entered centered.
- 2: Time values are entered aligned right.

1.3.18.21 TimeFormat Property

Defines the format of the time specification.

- 0: The information is provided in hh:mm
- -1: The information is provided in hh:mm:ss
- -2: The information is provided in hh:mm:ss.ms
- -3: The information is provided in hh:mm (full hours)
- -4: The information is provided in hh:mm:ss (full minutes)
- -5: The information is provided in hh:mm:ss.ms (full seconds)

1.3.18.22 TimeJump Property

WinCC Online Trend Control

The "Index" property references a trend. "TimeJump" defines whether the time jumps in the archive should be identified by the color defined in "TimeJumpColor".

WinCC Online Table Control

The value of this attribute cannot be changed. Read only access.

1.3.18.23 TimeJumpColor Property

WinCC Online Trend Control

The "Index" property references a trend. "TimeJumpColor" defines the color identifying the time jumps in the archive. Whether the information is evaluated is dependent on the value of the "TimeJump" property. The color is defines as an RGB value. LONG write-read access.

WinCC Online Table Control

The value of this property cannot be changed. Read only access.

1.3.18.24 TimeOverlap Property

WinCC Online Trend Control

The "Index" property references a trend. "TimeOverlap" defines whether the time overlaps in the archive should be identified by the color defined in "TimeOverlapColor".

WinCC Online Table Control

The value of this property cannot be changed. Read only access.

1.3.18.25 TimeOverlapColor Property

WinCC Online Trend Control

The "Index" property references a trend. "TimeOverlapColor" defines the color identifying the time overlaps in the archive. Whether the information is evaluated is dependent on the value of the "TimeOverlap" attribute. The color is defines as an RGB value.

WinCC Online Table Control

The value of this property cannot be changed. Read only access.

1.3.18.26 TimeRange Property

The "Index" property references a column pair or a trend. "TimeRange" defines how the time range to be displayed should be defined.

- 0: The time range to be displayed is defined by a starting time ("BeginTime") and stopping time ("EndTime").
- -1: The time range to be displayed is defined by a starting time ("BeginTime") and a time range ("TimeRangeBase" and "TimeRangeFactor").

1.3.18.27 TimeRangeBase Property

The "Index" property references a column pair or a trend. The time range to be displayed for this column pair/trend results from multiplying the values "TimeRangeBase" and "TimeRangeFactor", whereby the value "TimeRangeBase" is interpreted in milliseconds.

The "TimeRangeBase" and "TimeRangeFactor" properties are only evaluated when the "TimeRange" property is set, i.e. has the value "-1".

1.3.18.28 TimeRangeFactor Property

The "Index" property references a column pair or a trend. The time range to be displayed for this column pair/trend results from multiplying the values "TimeRangeBase" and "TimeRangeFactor", whereby the value "TimeRangeBase" is interpreted in milliseconds.

The "TimeRangeBase" and "TimeRangeFactor" properties are only evaluated when the "TimeRange" property is set, i.e. has the value "-1".

1.3.18.29 TimeStamp Property

Reads the time stamp of the last read access of a tag. The time stamp is returned in local time. DATE (read only)

The VBS standard function "FormatDateTime(Date[, NamedFormat])" enables the time stamp property to be output in plain text. The output is dependent on the current language setting. The language setting can be set using the VBS standard function SetLocale().

By implementing the second parameter of the FormatDate() function and further VBS standard functions such as Year, WeekDay, Day, Hour, Minute, Second enable the information, required by the user, to be split. Use the WeekdayName function to receive the name of the weekday for WeekDay.

Example:

```
'VBS87
Dim objTag
Dim lngCount
lngCount = 0
Set objTag = HMIRuntime.Tags("Tag11")
objTag.Read
SetLocale("en-gb")
MsgBox FormatDateTime(objTag.TimeStamp) 'Output: e.g. 06/08/2002 9:07:50
MsgBox Year(objTag.TimeStamp) 'Output: e.g. 2002
MsgBox Month(objTag.TimeStamp) 'Output: e.g. 8
MsgBox Weekday(objTag.TimeStamp) 'Output: e.g. 3
MsgBox WeekdayName(Weekday(objTag.TimeStamp)) 'Output: e.g. Tuesday
MsgBox Day(objTag.TimeStamp) 'Output: e.g. 6
MsgBox Hour(objTag.TimeStamp) 'Output: e.g. 9
MsgBox Minute(objTag.TimeStamp) 'Output: e.g. 7
MsgBox Second(objTag.TimeStamp) 'Output: e.g. 50
For lngCount = 0 To 4
MsgBox FormatDateTime(objTag.TimeStamp, lngCount)
Next
'lngCount = 0: Output: e.g. 06/08/2002 9:07:50
'lngCount = 1: Output: e.g. 06 August 2002
'lngCount = 2: Output: e.g. 06/08/2002
'lngCount = 3: Output: e.g. 9:07:50
```

Example 'IngCount = 4: Output: e.g. 9:07

The following example issues the time stamp of the tag "Tag1":

```
'VBS88
Dim objTag
Set objTag = HMIRuntime.Tags("Tag1")
objTag.Read
MsgBox objTag.TimeStamp
```

1.3.18.30 TimeZone Property

Defines the time zone used as a basis for displaying time values. Four settings are possible:

- Local time zone
- Server's time zone
- UTC (Universal Time Coordinated)
- Apply project settings (=> Use WinCC Explorer and access the computer's properties page to define the time mode specifically for the computer. The following are available for selection: WinCC V50 (Compatibility mode => Display as was standard in the individual display sections to V5), local time and UTC.

1.3.18.31 TitleCut Property

Defines whether the content of the fields of a title bar should be shortened if the column width is too small. Write/Read access.

1.3.18.32 Titleline Property

TRUE, when the control has a title bar and it can be moved in runtime. BOOLEAN write-read access.

1.3.18.33 Toggle Property

TRUE, when the round button should lock after being operated in runtime. BOOLEAN write-read access.

1.3.18.34 ToleranceHigh Property

Defines or returns the limit value for "Tolerance high".
The type of the evaluation (in percent or absolute) is defined in the TypeToleranceHigh property.
Monitoring of the limit value only takes effect when the CheckToleranceHigh property is set to "True".

1.3.18.35 ToleranceLow Property

Defines or returns the limit value for "Tolerance low".
The type of the evaluation (in percent or absolute) is defined in the TypeToleranceLow property.
Monitoring of the limit value only takes effect when the CheckToleranceLow property is set to "True".

1.3.18.36 ToolbarAlignment Property

Defines or returns the position of the toolbar. Write/Read access.

1.3.18.37 ToolbarButtons Property

Defines or returns the buttons contained in the toolbar by setting or resetting the corresponding bits. Each button is assigned to a bit. There are no restrictions concerning combination of the bits.

Bit - Value (hex) - Value (dec) - Button:

- 0 - 0x00000001 - 1 - Process message window
- 1 - 0x00000002 - 2 - Archive or:
- 2 - 0x00000004 - 4 - Archive
- 3 - 0x00000008 - 8 - Ackn. Central Signaling Devices
- 4 - 0x00000010 - 16 - Single acknowledgment
- 5 - 0x00000020 - 32 - Group acknowledgement
- 6 - 0x00000040 - 64 - Auto Scroll on/off

- 7 - 0x00000080 - 128 - Selection
- 8 - 0x00000100 - 256 - Set lock
- 9 - 0x00000200 - 512 - Protocol functions
- 11 - 0x00000800 - 2048 - Reset (not quit)
- 12 - 0x00001000 - 4096 - Start of list
- 13 - 0x00002000 - 8192 - End of list
- 14 - 0x00004000 - 16384 - Next message
- 15 - 0x00008000 - 32768 - Preview message
- 16 - 0x00010000 - 65536 - Info text
- 17 - 0x00020000 - 131072 - Comment (archive)
- 18 - 0x00040000 - 262144 - Loop-in-alarm
- 20 - 0x00100000 - 1048576 - Print
- 21 - 0x00200000 - 2097152 - Display Lock Window
- 22 - 0x00400000 - 4194304 - Lock/Unlock message
- 23 - 0x00800000 - 8388608 - Sort
- 24 - 0x01000000 - 16777216 - Time synchronization dialog

In order to display more buttons, their values must be logically linked with OR. Write/Read access.

1.3.18.38 Toolbar Property

TRUE, when a toolbar is to be displayed. BOOLEAN write-read access.

1.3.18.39 ToolbarHotKeys Property

Defines or returns hotkeys of the buttons in the toolbar. Write/Read access.

1.3.18.40 ToolTipText Property

Defines or returns the text to be displayed as a tooltip when the mouse is positioned over the object.

STRING (write-read access)

Example

The following example assigns a tooltip text to every object in the picture "NewPDL1": The picture "NewPDL1" comprises only objects containing the ToolTipText property:

```
'VBS89
Dim objScreen
Dim objScrltem
Dim lngIndex
Dim strName
lngIndex = 1
Set objScreen = HMIRuntime.Screens("NewPDL1")
For lngIndex = 1 To objScreen.ScreenItems.Count
  strName = objScreen.ScreenItems(lngIndex).ObjectName
  Set objScrltem = objScreen.ScreenItems(strName)
  '
  'Assign tooltip texts to the objects
  objScrltem.ToolTipText = "Name of object is " & strName
Next
```

1.3.18.41 Top Property

Function

Defines or returns the Y-coordinate of an object (measured from the top left edge of the picture) in pixels. The Y-coordinate relates to the top left corner of the rectangle enclosing the object.

LONG (write-read access)

Example

The following example shifts all objects in the picture "NewPDL1" 5 pixels upwards:

```
'VBS90
Dim objScreen
Dim objScrltem
Dim lngIndex
Dim strName
lngIndex = 1
Set objScreen = HMIRuntime.Screens("NewPDL1")
For lngIndex = 1 To objScreen.ScreenItems.Count
strName = objScreen.ScreenItems(lngIndex).ObjectName
Set objScrltem = objScreen.ScreenItems(strName)
objScrltem.Top = objScrltem.Top - 5
Next
```

1.3.18.42 TopConnectedConnectionPointIndex Property

Specifies or sets the index number of the top connecting point.

LONG write-read access.

1.3.18.43 TopConnectedObjectName Property

Specifies or sets the object name of the object which is docked on at the bottom connecting point.

LONG write-read access.

1.3.18.44 Transparent Property

TRUE, when the button appears completely filled in the color specified in "BackColor". BOOLEAN write-read access.

1.3.18.45 Trend Property

TRUE, when the tendency (rising or falling) of the measuring value being monitored should be displayed by a small arrow. BOOLEAN write-read access.

1.3.18.46 TrendColor Property

Defines or returns the color of the trend display.

The trend display indicates the tendency (rising or falling) of the measuring value being monitored by a small arrow. In order to activate the trend display, the Trend property must be set to "True". LONG write-read access.

1.3.18.47 Type Property

Reads out the object type, e.g. "Rectangle", "Circle" or "Line".

The object type is returned as a string. Read only

A special ID is returned as the type for all the graphic elements provided by WinCC. It can be found under the topic "Type Identification in VBS in the individual s of the WinCC Object Types.

Special feature

In the case of non-WinCC controls and OLE objects, the version-independent ProgID is returned as the type.

It is possible to determine the version-dependent ProgID or "User friendly Name" from the ProgID: In the following example, "Control1" is a control embedded in the picture which already returns the version-independent ProgID as a result of the Type property.

Note:

Since not every Control has a version-dependent ProgID, an error handling measure should be integrated to query the version-dependent ProgID or UserFriendlyName. If no error handling is used, the code is terminated immediately without any result when no ProgID is found.

Determine the version-dependent ProgID as follows:

```
'VBS91
Dim objControl
Dim strCurrentVersion
Set objControl = ScreenItems("Control1")
strCurrentVersion = CreateObject("WScript.Shell").RegRead("HKCR\" &
objControl.Type & "\CurVer")
MsgBox strCurrentVersion
```

Note:

In order that example above works, a multimedia control should be inserted in the picture.

Determine the User Friendly Name as follows:

```
'VBS92
Dim objControl
Dim strFriendlyName
Set objControl = ScreenItems("Control1")
strFriendlyName = CreateObject("WScript.Shell").RegRead("HKCR\" &
objControl.Type & "\")
```

```
MsgBox strFriendlyName
```

Note:

In order that example above works, a multimedia control should be inserted in the picture.

Example

The following example displays the type for all objects in the picture "NewPDL1":

```
'VBS93
Dim objScreen
Dim objScrltem
Dim lngIndex
Dim lngAnswer
Dim strName
lngIndex = 1
Set objScreen = HMIRuntime.Screens("NewPDL1")
For lngIndex = 1 To objScreen.ScreenItems.Count
strName = objScreen.ScreenItems(lngIndex).ObjectName
Set objScrltem = objScreen.ScreenItems(strName)
lngAnswer = MsgBox(objScrltem.Type, vbOKCancel)
If vbCancel = lngAnswer Then Exit For
Next
```

1.3.18.48 TypeAlarmHigh Property

TRUE, when the upper limit value, at which an alarm is triggered, should be evaluated as a percentage. FALSE, when the evaluation should be as an absolute value. BOOLEAN write-read access.

1.3.18.49 TypeAlarmLow Property

TRUE, when the lower limit value, at which an alarm is triggered, should be evaluated as a percentage. FALSE, when the evaluation should be as an absolute value. BOOLEAN write-read access.

1.3.18.50 TypeLimitHigh4 Property

TRUE, when the "Reserve 4" upper limit value should be evaluated as a percentage. FALSE, when the evaluation should be as an absolute value. BOOLEAN write-read access.

1.3.18.51 TypeLimitHigh5 Property

TRUE, when the "Reserve 5" upper limit value should be evaluated as a percentage. FALSE, when the evaluation should be as an absolute value. BOOLEAN write-read access.

1.3.18.52 TypeLimitLow4 Property

TRUE, when the "Reserve 4" lower limit value should be evaluated as a percentage. FALSE, when the evaluation should be as an absolute value. BOOLEAN write-read access.

1.3.18.53 TypeLimitLow5 Property

TRUE, when the "Reserve 5" lower limit value should be evaluated as a percentage. FALSE, when the evaluation should be as an absolute value. BOOLEAN write-read access.

1.3.18.54 TypeToleranceHigh Property

TRUE, when the "Tolerance high" lower limit value should be evaluated as a percentage. FALSE, when the evaluation should be as an absolute value. BOOLEAN write-read access.

1.3.18.55 TypeToleranceLow Property

TRUE, when the "Tolerance low" lower limit value should be evaluated as a percentage. FALSE, when the evaluation should be as an absolute value. BOOLEAN write-read access.

1.3.18.56 TypeWarningHigh Property

TRUE, when the "Warning high" lower limit value should be evaluated as a percentage. FALSE, when the evaluation should be as an absolute value. BOOLEAN write-read access.

1.3.18.57 TypeWarningLow Property

TRUE, when the "Warning low" lower limit value should be evaluated as a percentage. FALSE, when the evaluation should be as an absolute value.
BOOLEAN write-read access.

1.3.19 U

1.3.19.1 UnitColor Property

Defines the text color for the names of the unit of measurement. LONG write-read access.

1.3.19.2 UnitFont Property

Controls the display of the labeling for the unit of measurement. Read only access.

The following properties can be set:

- Font
- Font Style
- Font Size
- "Strikethrough" effect
- "Underline" effect

1.3.19.3 UnitOffset Property

This attribute defines the distance of the text for the unit of measurement in relation to the top edge of the object. The text can only be positioned along the vertical diameter of the graduated scale disk. The value of the property is related to the height of the object and is measured from the top edge of the object to the base of the text.

The value range is 0 to 1.

0: The base of the text is at the top limit of the object. The text is no longer visible because it is outside the object.

1: The base of the text is at the bottom limit of the object.

1.3.19.4 UnitText Property

Defines the text for the unit of measurement. Write/Read access.

1.3.19.5 UnselBGColor Property

Defines or returns the background color of entries in the text list object which are not selected. LONG write-read access.

1.3.19.6 UnselTextColor Property

Defines or returns the color of the text for entries in the text list object which are not selected. LONG write-read access.

1.3.19.7 UserValue1 Property

Defines or returns a value.

The value can be evaluated by a script, for example. This value is neither evaluated nor displayed in runtime.

1.3.19.8 UserValue2 Property

Defines or returns a value.

The value can be evaluated by a script, for example. This value is neither evaluated nor displayed in runtime.

1.3.19.9 UserValue3 Property

Defines or returns a value.

The value can be evaluated by a script, for example. This value is neither evaluated nor displayed in runtime.

1.3.19.10 UserValue4 Property

Defines or returns a value.

The value can be evaluated by a script, for example. This value is neither evaluated nor displayed in runtime.

1.3.19.11 UpdateCycle Property

Returns the type and frequency of updating the picture window in runtime. Read only access.

1.3.19.12 UpperLimit Property

TRUE, when the "UpperLimitColor" specification is to be used in order to identify the tag values (from a trend referenced via "Index") which lie above the value defined in "UpperLimitValue". BOOLEAN write-read access.

1.3.19.13 UpperLimitColor Property

Defines the color to be used in order to identify the tag values (from a trend referenced via "Index") which lie above the value defined in "UpperLimitValue". Whether the information is evaluated is dependent on the value of the "UpperLimit" property. The color is defines as an RGB value. LONG write-read access.

1.3.19.14 UpperLimitValue Property

Tag values (from a trend referenced via "Index") which lie above the value defined by "UpperLimitValue" are identified by the color specified in "UpperLimitColor". Whether the information is evaluated is dependent on the value of the "UpperLimit" property.

1.3.20 V-Z

1.3.20.1 ValueColumnAlignment Property

The "Index" property references a pair of columns. "ValueColumnAlignment" defines the alignment of the tag value for this column pair.

- 0: Tag values are entered aligned left.
- 1: Tag values are entered centered.
- 2: Tag values are entered aligned right.

1.3.20.2 Value Property

Displays the value of the tags at the last read access or the value which should be or was written. Value stands for the value of a tag. After calling in the "Read" method, the tag value read is returned. Before writing, the new tag value required can be assigned to the property. After calling in the "Write" method, the property contains the value last written.

VARIANT (write-read access)

WinCC Gauge Control

Defines the value to which the pointer points. Value range: "ValueMin" to "ValueMax".

Example

The following example writes a new value in the "Tag1" tag:

```
'VBS94
Dim objTag
Set objTag = HMIRuntime.Tags("Tag1")
objTag.Value = 50
objTag.Write
```

1.3.20.3 ValueMax Property

Defines the value at the end of the scale. Write/Read access.

1.3.20.4 ValueMin Property

Defines the value at the start of the scale. Write/Read access.

1.3.20.5 Variable Property

The "Index" property references a pair of columns. "Tag" defines the name of the tag which should be connected to this column pair.

1.3.20.6 Visible Property

witches an object visible or invisible or issues a corresponding value:

- TRUE: Object is visible
- FALSE: Object is invisible

VARIANT_BOOL (write-read access)

Example

The following example sets all the objects in the picture "NewPDL1" to invisible:

```
'VBS95
Dim objScreen
Dim objScrltem
Dim lngIndex
Dim strName
lngIndex = 1
Set objScreen = HMIRuntime.Screens("NewPDL1")
For lngIndex = 1 To objScreen.ScreenItems.Count
strName = objScreen.ScreenItems(lngIndex).ObjectName
Set objScrltem = objScreen.ScreenItems(strName)
objScrltem.Visible = False
Next
```

1.3.20.7 Warning Property

Defines the start of the "Warning zone" as a scale value. Write/Read access.

1.3.20.8 WarningColor Property

Defines the color of the "Warning zone" o the scale. LONG write-read access.

1.3.20.9 WarningHigh Property

Defines or returns the upper limit value for "Warning High".

In order that the limit value is monitored, the "CheckWarningHigh" property must be set to TRUE.

The display on reaching the limit value and the type of evaluation are defined by means of the "ColorWarningHigh" and "TypeWarningHigh" properties.

1.3.20.10 WarningLow Property

Defines or returns the lower limit value for "Warning Low".

In order that the limit value is monitored, the "CheckWarningLow" property must be set to TRUE.

The display on reaching the limit value and the type of evaluation are defined by means of the "ColorWarningLow" and "TypeWarningLow" properties.

1.3.20.11 Width Property

Sets or outputs the width of an object in pixels.

LONG

Example

The following example doubles the width of all objects in the pictures "NewPDL1" whose name begins with "Button":

```
'VBS96
Dim objScreen
Dim cmdButton
Dim lngIndex
Dim strName
lngIndex = 1
Set objScreen = HMIRuntime.Screens("NewPDL1")
For lngIndex = 1 To objScreen.ScreenItems.Count
    'Get all "Buttons"
    strName = objScreen.ScreenItems(lngIndex).ObjectName
    If "Button" = Left(strName, 6) Then
        Set cmdButton = objScreen.ScreenItems(strName)
        cmdButton.Width = cmdButton.Width * 2
    End If
Next
```

1.3.20.12 WindowBorder Property

TRUE, when the window is displayed with borders in runtime. Read only access.

1.3.20.13 WindowsStyle Property

TRUE, when the object complies with the general Windows style (e.g. gray buttons without borders). BOOLEAN write-read access. Note:

- When this property is set to "True", the properties which do not comply with the Windows style are ignored (e.g. "BorderWidth").
- On the other hand, the definition of a "BorderWidth" or a background color other than gray causes "WindowsStyle" to receive the value "False".
- Exceptions here are the flash attributes: The definition of flash attributes does not automatically lead to the deactivation of the "WindowsStyle" attribute.

1.3.20.14 WindowType Property

Defines the use of the message window.

- 0 - Message Window: For displaying the currently queued messages.
- 1 - Archive Window: For displaying the archived messages, or:
- 2 - Archive Window: For displaying the archived messages.
- 3 - Lock List: For displaying the currently locked messages.

1.3.20.15 WithAxes Property

TRUE, when the scale should be displayed. BOOLEAN write-read access.

1.3.20.16 WithLabels Property

TRUE, when the scale labels should be displayed. BOOLEAN write-read access.

1.3.20.17 ZeroPoint Property

Defines or returns the position of the zero point of the bar graph.

Specify the value as a %age of the total bar height. The zero point can also be outside of the range represented.

The "ScalingType" property must be set to "2" and "Scaling" to TRUE.

1.3.20.18 ZeroPointValue Property

Defines the value of the zero point of the scale indicator.

Defines or returns the absolute value for the zero point.

1.3.20.19 Zoom Property

Sets the zoom level within a picture or reads it out.

Example

The following example doubles the zoom level of the current picture:

```
'VBS97
HMIRuntime.ActiveScreen.Zoom = HMIRuntime.ActiveScreen.Zoom * 2
```

1.3.20.20 Zoomable Property

TRUE, when the display of the DXF picture in Runtime mode supports the zooming function. BOOLEAN write-read access.

1.4 Methods

Overview

Methods, which are applied to individual objects, can be used to read out tag values for further processing or displaying diagnostics messages in runtime.

1.4.1 Activate Method

Function

Activates the picture specified.

Syntax

Ausdruck.**Activate**

Expression

Necessary. An expression which returns a "Screen" type object.

Parameter

--

Example

The following indicates its use:

```
'VBS98
Dim objScreen
MsgBox HMIRuntime.ActiveScreen.ObjectName 'Output of active screen
Set objScreen = HMIRuntime.Screens("ScreenWindow1")
objScreen.Activate 'Activate "ScreenWindow1"
MsgBox HMIRuntime.ActiveScreen.ObjectName 'New output of active screen
```

1.4.2 Item Method

Function

Retrieves an object from a collection and enables access to it via Index.

Syntax

Ausdruck.Item()

Expression

Necessary. An expression which returns an object of the type "Screens", "Layers" (or "Tags").

Caution

In the case of "Tags", restricted functional scope! The standard methods get_Count and get_NewEnum are missing so that access via Index nor the counting of all tags is possible.

Parameter

VARIANT

Example

The following example issues the names of all the objects contained in the picture "NewPDL1":

```
'VBS99
Dim objScreen
Dim objScrItem
Dim lngIndex
Dim lngAnswer
Dim strName
lngIndex = 1
Set objScreen = HMIRuntime.Screens("NewPDL1")
For lngIndex = 1 To objScreen.ScreenItems.Count
    'The objects will be indicate by Item()
    strName = objScreen.ScreenItems.Item(lngIndex).ObjectName
    Set objScrItem = objScreen.ScreenItems(strName)
    lngAnswer = MsgBox(objScrItem.ObjectName, vbOKCancel)
    If vbCancel = lngAnswer Then Exit For
Next
```

1.4.3 Read Method

Function

Reads out the stats of a tag (tag object) shortly after the moment it was called. At the same time, the tag object is provided with the values read. On reading the tag, the value, quality code and time stamp of the tag are established. The "LastError" property can be used to determine whether the call was successful.

The "Name", "ServerPrefix" and "TagPrefix" properties are not changed as a result.

If the value of the tag is read successfully, the properties of the tag object are assigned the following values:

Value	Tag values
Name	Tag name (unchanged)
QualityCode	Quality level
TimeStamp	Current tag time stamp
LastError	0
Error	" "

If the value of the tag is not read successfully, the properties of the tag object are assigned the following values:

Value	VT_Empty
Name	Tag name (unchanged)
QualityCode	BAD NON-SPECIFIC
TimeStamp	0
LastError	Read operation error codes
Error	Error on LastError

Note:

A summary of the possible Quality Codes is provided in the WinCC Information System under the heading "Communication" > "Diagnostics" or "Communication" > "Quality Codes".

Syntax

Ausdruck.**Read**([Readmode])

Expression

Necessary. An expression which returns a tag object. The return value of the Read method is the value of the tag read out.

Parameter

The optional "Readmode" parameter enables the distinction between two types of reading:

Parameter	
0	The tag value is read from the process image (cache). 0 is the default value.

Parameter	
1	The value is read directly from the PLC or channel (direct).

If the "Readmode" parameter is omitted, the value is read from the process image by default. The return value of the Read method is the tag value read out as VARIANT.

Possible error codes (LastError)

The following error codes are defined:

Code in hexadecimal notation	
0x00000000	OK
0x80040001	Error in execution
0x80040002	Tag error

Possible error (Error)

Output	
(" ")	OK
("Operation failed")	Execution error
("Variable not found")	Tag error

Reading From the Process Image

When reading from the process image, the tag is logged on and, from that moment, polled cyclically from the PLC. The login cycle is dependent on the configured trigger. The value is read by WinCC from the tag image. On closing the picture, the tags are logged off again. The call is characterized by the following:

- The value is read by WinCC from the tag image.
- The call is faster in comparison to direct reading (except with the first call: The first call basically takes longer because the value from the PLC must be read out and logged on.)
- The duration of the call is not dependent on the bus load or AS.

Behavior in actions with a tag trigger

All the tags contained in the tag trigger are already known on opening a picture and are logged in with the specified monitoring interval. Since all tags can be requested in one operation, the best possible optimization can be achieved by the channel. If a tag, contained in the trigger, is requested with Read during an action, the value already exists and is transferred directly to the call. If a tag is requested which is not contained in the trigger, the behavior is the same as with a standard trigger.

Behavior in actions with a cyclic trigger

When called for the first time, the tag is logged on with half the cycle time. The value is then available for each subsequent call.

Behavior in event-driven actions

When called for the first time, the tag is logged on in "Upon Change" mode. Process tags, logged on in "Upon Change" mode, correspond to a cyclic read job with a cycle time of 1s.

If an event (e.g. mouse click) requests a value asynchronously, the tag is transferred to the tag image. From this moment, the tag is polled cyclically from the AS, thus increasing the basic load. To bypass this increase in the basic load, the value can also be read synchronously. The synchronous call causes a one-off increase in the communication load but the tag is not transferred to the tag image.

Direct reading

In the case of direct reading, the current value is returned. The tag is not logged on cyclically but the value is requested once from the AS. Direct reading has the following properties:

- The value is read explicitly from the AS.
- The call takes longer compared to reading from the process image.
- The duration of the call is dependent on the bus load and AS, amongst other things.

Example

Reading a tag directly from the PLC or channel

```
'VBS100
Dim objTag
Dim vntValue
Set objTag = HMIRuntime.Tags("Tagname")
vntValue = objTag.Read(1) 'Read direct
MsgBox vntValue
```

Reading a tag from the process image

```
'VBS101
Dim objTag
Dim vntValue
Set objTag = HMIRuntime.Tags("Tagname")
vntValue = objTag.Read 'Read from cache
MsgBox vntValue
```

1.4.4 Refresh Method

Function

Drawing all visible pictures again.

Syntax

Ausdruck.**Refresh**
Expression

Necessary. An expression which returns a "Screens" or "Screen" type object.

Parameter

--

Examples

The first example forces all visible pictures to be drawn again:

```
'VBS149  
HMIRuntime.Screens.Refresh
```

The second example forces the basic picture to be immediately redrawn:

```
'VBS150  
HMIRuntime.Screens(1).Refresh
```

1.4.5 Stop Method

Function

Terminates WinCC Runtime.

Syntax

HMIRuntime.**Stop**

Parameter

Example

The following example terminates WinCC runtime:

```
'VBS124  
HMIRuntime.Stop
```

1.4.6 Trace Method

Displays messages in the dialog window.

Syntax

HMIRuntime.**Trace**

Parameter

STRING

Example

The following example writes a text in the diagnostics window:

```
'VBS103  
HMIRuntime.Trace "Customized error message"
```

1.4.7 Write Method

Function

Writes a value synchronously or asynchronously in a tag. The "LastError" property can be used to determine whether the call was successful.

If the value of the tag is set successfully, the properties of the tag object are assigned the following values:

Value	Tag values set by the user (unchanged)
Name	Tag name (unchanged)
QualityCode	BAD NON-SPECIFIC
TimeStamp	0
LastError	0
Error	" "

If the value of the tag is not set successfully, the properties of the tag object are assigned the following values:

Value	Tag values set by the user (unchanged)
Name	Tag name (unchanged)
QualityCode	BAD NON-SPECIFIC
TimeStamp	0
LastError	Write operation error codes
Error	Error on LastError

Syntax

Ausdruck.**Write** [Value],[Writemode]

Expression

Necessary. An expression which returns a tag object.

Parameter

The value to be written can be transferred directly to the method as a parameter. If the parameter is not specified, the value in the "Value" property is used. The "Writemode" option parameter can be used to select whether the tag value should be written synchronously or asynchronously. If the "Writemode" parameter is not used, writing is performed asynchronously as its default value.

During the writing process, no information is supplied on the status of the tags.

The "Value" property contains the value which was set before or during the writing operation, therefore it may not correspond to the real current value of the tag. If the data on the tag should be updated, use the Read method.

Parameter	
Value (optional)	<p>The tag value is specified. The specified value overwrites the value in the "Value" property in the tag object.</p> <p>The tag value is not specified. The tag receives the current value from the "Value" property of the tag object.</p>
Writemode (optional)	<p>0 or empty: The tag value is written asynchronously. 0 is the default value.</p> <p>1: The tag value is written synchronously.</p>

On asynchronous writing, it is written immediately into the tag image. The user does not receive any feedback if the value has been written in the programmable controller, too.

In the case of synchronous writing (direct to the PLC), the writing operation actually occurs when the PLC is ready to operate. The user receives a check-back message if the writing operation was not successful.

Possible error codes (LastError)

The following error codes are defined:

Code in hexadecimal notation	
0x00000000	OK
0x80040001	Error in execution
0x80040002	Tag error

Possible error (Error)

Output	
(" ")	OK
("Operation failed")	Execution error
("Variable not found")	Tag error

Example

Asynchronous writing

```
'VBS104  
Dim objTag  
Set objTag = HMIRuntime.Tags("Var1")  
objTag.Value = 5  
objTag.Write  
MsgBox objTag.Value
```

or

```
'VBS105  
Dim objTag  
Set objTag = HMIRuntime.Tags("Var1")  
objTag.Write 5  
MsgBox objTag.Value
```

Synchronous writing

```
'VBS106  
Dim objTag  
Set objTag = HMIRuntime.Tags("Var1")  
objTag.Value = 5  
objTag.Write ,1  
MsgBox objTag.Value
```

or

```
'VBS107  
Dim objTag  
Set objTag = HMIRuntime.Tags("Var1")  
objTag.Write 5, 1  
MsgBox objTag.Value
```

|

Index

3			
3D bar	29		
A			
ActiveScreen object	3		
ActiveScreenItem object	3		
Alarm control	46		
Analog Clock	47		
Application window	30		
B			
Bar	31		
Button	40		
C			
Check box	41		
Circle	20		
Circular arc	20		
Connector	28		
Control	33		
Controls	45		
HMI Symbol Library 1.3	54		
WinCC Alarm Control	46		
WinCC Digital Analog Clock	47		
WinCC DXF Control	48		
WinCC Function Trend Control	48		
WinCC Online Table Control	49		
WinCC Online Trend Control	50		
WinCC Push Button Control	51		
WinCC Slider Control	53		
Customized object	55		
D			
Digital Analog Clock	47		
Digital Clock	47		
DXF Control	48		
E			
Ellipse	18		
Ellipse arc	18		
		Ellipse segment	19
		Error codes	107
		Error messages	90
F			
		Function Trend Control	48
G			
		Gauge Control	49
		Graphic object	35
		property	17
		types in VBS	17
		WinCC Gauge Control	49
		Graphic objects	17
		Group	55
		Group display	38
H			
		HMI Symbol Library 1.3	54
		HMIRuntime object	4
I			
		I/O field	35
		Item object	4
L			
		Language-IDs	106
		Layer object	5
		Layers object (listing)	6
		Line	22
		Lists	2
		layers	6
		ScreenItems	10
		Screens	13
		tags	16
M			
		Methods	172
		activate	172
		item	173
		read	174

Refresh	177	AlarmLow	59
stop	177	alignment	60
trace	178	AlignmentTop	60
write	178	AllowPersistence	60, 73
O			
Object model	1	AllServer	60
lists	2	Analog	61
methods	172	AngleAlpha	61
objects	2	AngleBeta	61
properties	56	AngleMax	61
Object types	17	AngleMin	61
Controls	45	application	61
customized object	55	archive	62
group	55	assignments	62
ScreenItem	17	AssumeOnExit	62
smart objects	28	AssumeOnFull	62
standard objects	18	AutoRange	62
Windows objects	40	AutoRangeX	63
Objects	2	AutoRangeY	63
ActiveScreen	3	AutoScroll	63
ActiveScreenItem	3	AutoSize	63
HMIRuntime	4	average	64
Item	4	axe	64
layer	5	AxisSection	64
parent	6	BackBorderWidth	64
screen	11	BackColor	64
ScreenItem	8	BackColor2	65
Tag	14	BackColor3	65
OLE object	36	BackColorBottom	65
Online Table Control	49	BackColorTop	65
Online Trend Control	50	BackFlashColorOff	66
Option box	42	BackFlashColorOn	66
P			
Parent object	6	Background	66
Picture Window	32	BackgroundPicture	66
Pie segment	21	BackStyle	66
Polygon	22	BarBackColor	67
Polyline	23	BarDepth	67
Properties in VBS	56	BarFillColor	67
AccessPath	57	BarHeight	67
activate	57	BarWidth	67
ActiveScreen	57	BasePicReferenced	67
ActiveScreenItem	57	BasePicTransparentColor	68
Actualize	58	BasePicture	68
ActualPointLeft	58	BasePicUseTransparentColor	68
ActualPointTop	58	BaseScreenName	68
AdaptPicture	59	BaseX	69
AdaptSize	59	BaseY	69
AlarmHigh	59	BeginTime	69
		BeginValue	69
		BeginX	69
		BeginY	69
		BevelColorDown	70
		BevelColorUp	70
		BevelInner	70
		BevelOuter	70
		BevelWidth	71

BitNumber	71	ColorAlarmHigh	81
BlinkColor	71	ColorAlarmLow	81
BlinkMode	71	ColorBottom	81
BlinkSpeed	71	ColorChangeType	81
BorderBackColor	71	ColorLimitHigh4	81
BorderColor	72	ColorLimitHigh5	81
BorderColorBottom	72	ColorLimitLow4	82
BorderColorTop	72	ColorLimitLow5	82
BorderEndStyle	72	ColorToleranceHigh	82
BorderFlashColorOff	72	ColorToleranceLow	82
BorderFlashColorOn	72	ColorTop	82
BorderStyle	72	ColorWarningHigh	82
BorderWidth	73	ColorWarningLow	83
BottomConnectedConnectionPointIndex	73	ColTitle	83
	73	ColWidth	83
BottomConnectedObjectName	73	Command	83
BoxAlignment	73	CommonTime	83
BoxType	73	CommonX	83
Button1Width	75	CommonY	83
Button2Width	75	ContinousChange	84
Button3Width	75	CurrentContext	84
Button4Width	75	Cursor	84
ButtonColor	74	CursorControl	84
ButtonCommand	74	CurveForm	85
Caption	75	Danger	85
CaptionColor	76	DangerColor	85
CaptionFont	76	DataFormat	85
CaptionOffset	76	DataIndex	86
CaptionText	76	DataX	86
CellCut	76	DataXY	86
CenterColor	76	DataY	86
CenterScale	77	DeleteData	86
CheckAlarmHigh	77	Delta	86
CheckAlarmLow	77	DesiredCurveColor	86
CheckLimitHigh4	77	DesiredCurveCurveForm	87
CheckLimitHigh5	77	DesiredCurveNumberOfUAValues	87
CheckLimitLow4	78	DesiredCurveSourceUAArchive	87
CheckLimitLow5	78	DesiredCurveSourceUAArchiveStartID	87
CheckToleranceHigh	78	DesiredCurveSourceUAColumnX	87
CheckToleranceLow	78	DesiredCurveSourceUAColumnY	88
CheckWarningHigh	78	DesiredCurveVisible	88
CheckWarningLow	79	Direction	88
ClearOnError	79	DXFFilename	88
ClearOnNew	79	Edit	89
Closeable	79	Editable	89
CloseButton	79	EditAtOnce	89
CoarseGrid	79	Enabled	89
CoarseGridValue	80	EndAngle	90
CoarseGridValueX	80	EndTime	90
CoarseGridValueY	80	EndValue	90
CoarseGridX	79	EndX	90
CoarseGridY	80	EndY	90
CollectValue	80	ErrorDescription	90
ColMove	80	Exponent	91
Color	80	ExtendedOperation	91

FillColor	91	HandFillColor	101
Filling	92	Handtype	102
FillingIndex	92	Height	102
FillStyle	92	HiddenInput	102
FillStyle2	93	Hotkey	103
FineGrid	93	HourNeedleHeight	103
FineGridValue	93	HourNeedleWidth	103
FineGridValueX	93	Hysteresis	103
FineGridValueY	94	HysteresisRange	103
FineGridX	94	Index	104
FineGridY	94	InnerBevelOffset	104
FlashBackColor	94	InnerBevelStyle	104
FlashBorderColor	94	InnerBevelWidth	104
FlashFlashPicture	94	InsertData	105
FlashForeColor	94	ItemBorderBackColor	105
FlashPicReferenced	94	ItemBorderColor	105
FlashPicTransColor	95	ItemBorderStyle	105
FlashPicture	95	ItemBorderWidth	105
FlashPicUseTransColor	95	ItemVisible	105
FlashRate	95	Label	106
FlashRateBackColor	95	LabelColor	106
FlashRateBorderColor	96	LabelX	106
FlashRateFlashPic	96	LabelY	106
FlashRateForeColor	96	Language	106
Flip	96	LanguageSwitch	106
FocusColor	96	LastError	107
FocusRect	97	Layer	107
FocusWidth	97	Layer00Checked	108
Font	97	Layer00Color	110
FontBold	97	Layer00Value	112
FontItalic	98	Layer01Checked	108
FontName	98	Layer01Color	110
FontPosition	98	Layer01Value	112
FontSize	98	Layer02Checked	108
FontStrikeThru	98	Layer02Color	110
FontUnderline	98	Layer02Value	112
ForeColor	98	Layer03Checked	108
ForeFlashColorOff	99	Layer03Color	110
ForeFlashColorOn	99	Layer03Value	112
FrameColor	99	Layer04Checked	109
FrameColorDown	99	Layer04Color	111
FrameColorUp	99	Layer04Value	112
FramePicture	99	Layer05Checked	109
FrameScale	99	Layer05Color	111
FrameWidth	100	Layer05Value	113
FreezeProviderConnections	100	Layer06Checked	109
GraphDirection	100	Layer06Color	111
GridLineHorz	100	Layer06Value	113
GridLines	100	Layer07Checked	109
GridLinesValueX	101	Layer07Color	111
GridLinesValueY	101	Layer07Value	113
GridLinesX	101	Layer08Checked	109
GridLinesY	101	Layer08Color	111
GridLineValue	101	Layer08Value	113
GridLineVert	101	Layer09Checked	110

Layer09Color	111	Min	122
Layer09Value	113	MinuteNeedleHeight	122
Layer10Checked	110	MinuteNeedleWidth	122
Layer10Color	112	Movable	122
Layer10Value	113	MsgCtrlFlags	123
Left	114	MsgFilterSQL	123
LeftComma	114	Name	123
LightEffect	114	NeedleColor	124
LimitHigh4	114	NormalColor	124
LimitHigh5	115	NumberLines	124
LimitLow4	115	NumItems	124
LimitLow5	115	Object	124
LimitMax	115	ObjectName	125
LimitMin	115	OffsetLeft	125
LineFont	115	OffsetTop	125
LineHeight	116	Online	126
LineTitle	116	OnTop	126
ListType	116	OperationMessage	126
LoadDataImmediately	116	OperationReport	126
LocaleID	116	Orientation	127
LockBackColor	116	OuterBevelStyle	127
LockStatus	117	OuterBevelWidth	127
LockText	117	Outline	127
LockTextColor	117	OutputFormat	127
LongStrokesBold	117	OutputValue	127
LongStrokesOnly	117	Parent	128
LongStrokesSize	117	PasswordLevel	128
LongStrokesTextEach	117	PersistentRT	128
LowerLimit	118	PersistentRTCS	128
LowerLimitColor	118	PersistentRTCSPermission	128
LowerLimitValue	118	PersistentRTPermission	129
Marker	118	PicDeactReferenced	129
Max	119	PicDeactTransparent	129
MaximizeButton	119	PicDeactUseTransColor	129
MCGUBackColorOff	119	PicDownReferenced	129
MCGUBackColorOn	119	PicDownTransparent	129
MCGUBackFlash	119	PicDownUseTransColor	130
MCGUTextColorOff	119	PicReferenced	130
MCGUTextColorOn	119	PicTransColor	130
MCGUTextFlash	120	Picture	130
MCKOBackColorOff	120	PictureBack	130
MCKOBackColorOn	120	PictureDeactivated	130
MCKOBackFlash	120	PictureDown	130
MCKOTextColorOff	120	PictureName	131
MCKOTextColorOn	120	PictureSelected	131
MCKOTextFlash	120	PictureThumb	131
MCKQBackColorOff	121	PictureUnselected	131
MCKQBackColorOn	121	PictureUp	131
MCKQBackFlash	121	PicUpReferenced	131
MCKQTextColorOff	121	PicUpTransparent	131
MCKQTextColorOn	121	PicUpUseTransColor	132
MCKQTextFlash	121	PicUseTransColor	132
MCText	121	PointCount	134
MeasurePoints	122	Position	132
MessageClass	122	Precisions	132

PrecisionX	132	ShowPeak	143
PrecisionY	132	ShowPosition	144
PredefinedAngles	133	ShowRulerImmediately	144
Pressed	133	ShowThumb	144
PrintJob	133	ShowValuesExponentialX	144
Process	133	ShowValuesExponentialY	144
ProjectPath	133	ShowWarning	144
ProviderClsID	133	SignificantMask	145
ProviderType	134	Sizeable	145
QualityCode	134	SmallChange	145
Radius	135	SourceBeginTime	145
RadiusHeight	135	SourceEndTime	145
RadiusWidth	135	SourceNumberOfUAValues	146
RangeMax	135	SourceNumberOfValues	146
RangeMin	135	SourceTagNameX	146
Rectangular	135	SourceTagNameY	146
ReferenceRotationLeft	136	SourceTagProviderDataX	147
ReferenceRotationTop	136	SourceTagProviderDataY	147
RelayCurves	136	SourceTimeRange	147
Relevant	136	SourceUAArchive	147
Replacement	136	SourceUAArchiveStartID	148
ReplacementColor	136	SourceUAColumnX	148
RightComma	137	SourceUAColumnY	148
Rotation	137	SquareExtent	148
RotationAngle	137	StartAngle	148
RoundCornerHeight	137	Statusbar	149
RoundCornerWidth	137	StatusbarPanels	149
RulerPrecisions	137	StatusbarStretch	149
RulerPrecisionX	138	TagName	149
RulerPrecisionY	138	TagPrefix	150
SameSize	138	TagProviderClsid	150
ScaleColor	138	Tags	150
ScaleTicks	138	Template	150
Scaling	138	Text	151
ScalingType	139	ThumbBackColor	151
ScalingTypeX	139	TicColor	151
ScalingTypeY	139	TicFont	151
Screen	139	Ticks	153
ScreenItems	140	TicksColor	153
ScreenName	140	TickStyle	153
Screens	140	TicOffset	152
Scrollable	140	TicTextColor	152
ScrollBars	141	TicTextOffset	152
SecondNeedleHeight	141	TicWidth	152
SecondNeedleWidth	141	TimeAxis	153
SelBGColor	141	TimeAxisFormat	153
SelectionMode	142	TimeAxisX	154
SelTextColor	142	TimeColumnAlignment	154
ServerData	142	TimeFormat	154
ServerNames	142	TimeJump	154
ServerPrefix	142	TimeJumpColor	155
ShowBar	143	TimeOverlap	155
ShowDanger	143	TimeOverlapColor	155
ShowDecimalPoint	143	TimeRange	155
ShowNormal	143	TimeRangeBase	156

rounded rectangles	25	V	
Slider	43	VBS	
static text	27	lists	2
status display	39	methods	172
text list	38	object model	1
WinCC Alarm Control	46	object types	17
WinCC Digital Analog Clock	47	objects	2
WinCC DXF Control	48	properties	56
WinCC Function Trend Control	48	reference	1
WinCC Gauge Control	49		
WinCC Online Table Control	49	W	
WinCC Online Trend Control	50	WinCC	
WinCC Push Button Control	51	graphic object types	17
WinCC Slider Control	53	WinCC Alarm Control	46
ScreenItem object	8	WinCC Digital Analog Clock	47
ScreenItems object (listing)	10	WinCC DXF Control	48
Screens object (listing)	13	WinCC Function Trend Control	48
Slider	43	WinCC Gauge Control	49
Slider Control	53	WinCC Online Table Control	49
Smart objects	28	WinCC Online Trend Control	50
3D bar	29	WinCC Push Button Control	51
application window	30	WinCC Slider Control	53
bar	31	Windows objects	40
control	33	button	40
graphic object	36	check box	41
group display	38	option box	42
I/O field	35	round button	42
OLE object	36	slider	43
picture window	32		
status display	39		
text list	38		
Standard objects	18		
circle	20		
circular arc	20		
connector	28		
ellipse	18		
ellipse arc	18		
ellipse segment	19		
line	22		
pie segment	21		
polygon	22		
polyline	23		
rectangle	24		
rounded rectangles	25		
static text	27		
Static text	27		
Status Display	39		
Symbol Library	54		
T			
Tag object	14		
Tags object (listing)	16		
Text list	38		