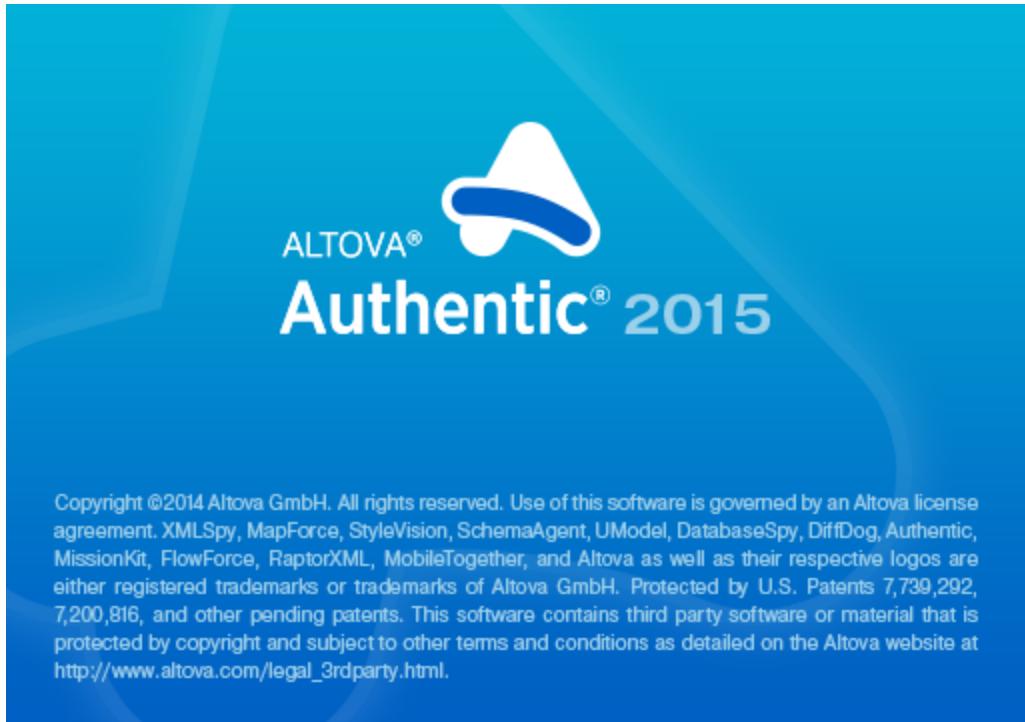


Manual del usuario y de referencia



Copyright ©2014 Altova GmbH. All rights reserved. Use of this software is governed by an Altova license agreement. XMLSpy, MapForce, StyleVision, SchemaAgent, UModel, DatabaseSpy, DiffDog, Authentic, MissionKit, FlowForce, RaptorXML, MobileTogether, and Altova as well as their respective logos are either registered trademarks or trademarks of Altova GmbH. Protected by U.S. Patents 7,739,292, 7,200,816, and other pending patents. This software contains third party software or material that is protected by copyright and subject to other terms and conditions as detailed on the Altova website at http://www.altova.com/legal_3rdparty.html.

Manual del usuario y de referencia de Altova Authentic 2015 Browser Edition

Todos los derechos reservados. Ningún fragmento de esta publicación podrá ser reproducido de manera alguna (ya sea de forma gráfica, electrónica o mecánica, fotocopiado, grabado o reproducido en sistemas de almacenamiento y recuperación de información) sin el consentimiento expreso por escrito de su autor/editor.

Los productos a los que se hace referencia en este documento pueden ser marcas registradas de sus respectivos propietarios. El autor y editor no afirman ser propietarios de dichas marcas registradas.

Durante la elaboración de este documento se tomaron todas las precauciones necesarias para prevenir errores. Sin embargo, el autor y editor no se responsabilizan de los errores u omisiones que pudiese contener el documento ni de los posibles daños o perjuicios derivados del uso del contenido de este documento o de los programas y código fuente que vengan con el documento. Bajo ninguna circunstancia se podrá considerar al autor y editor responsables de la pérdida de beneficios ni de cualquier otro daño y perjuicio derivado directa o indirectamente del uso de este documento.

Fecha de publicación: 2014

© 2014 Altova GmbH

Tabla de contenido

1	Altova Authentic Browser Edition	2
2	Información sobre este manual	4
3	Introducción	6
3.1	Ventajas de la edición Authentic Browser	7
3.2	Funcionamiento	8
3.3	. Versiones de la edición Authentic Browser	10
4	Configuración del servidor	13
4.1	.IIS: configurar el servicio de explorador	15
4.2	.Archivos XSD, XML y SPS/PXF	17
4.3	.Página HTML para el complemento de Authentic	19
4.3.1	Licencia para la edición Enterprise Edition	21
4.3.2	Internet Explorer	23
–	El elemento OBJECT	24
–	El elemento SCRIPT	28
–	Ejemplo nº1: ejemplo básico.....	29
–	Ejemplo nº2: ordenar una tabla.....	31
4.3.3	Firefox	34
–	El elemento EMBED	35
–	Añadir agentes de escucha de eventos.....	38
–	Ejemplo nº1: ejemplo básico.....	39
–	Ejemplo nº2: ordenar una tabla.....	41
4.3.4	Para cualquier explorador	44
–	Ejemplo para cualquier explorador.....	45
4.4	.Paquetes de extensión para instalación por encargo	50
5	Configuración de clientes	52
5.1	.Requisitos	53
5.2	.Complemento para exploradores Authentic Browser	54

5.2.1	Instalación por encargo	55
5.2.2	Instalación manual con los paquetes de extensión	56
5.2.3	Instalación manual con el archivo de instalación MSI	57
5.2.4	Instalación de inserción con el archivo de instalación MSI	58
5.2.5	Actualizaciones automáticas	59
5.2.6	Desinstalación y deshabilitación	60
5.3	.Configuración de seguridad en IE9	61
5.4	.Configuración de seguridad en IE10	63

6 Referencia 65

6.1	.Mecanismos	66
6.1.1	Eventos: punto de conexión (para IE)	67
6.1.2	Eventos: añadir agentes de escucha de eventos (para Firefox)	69
6.1.3	Eventos: botones de las barras de herramientas	70
6.1.4	Eventos: referencia	71
6.1.5	Acceder al contenido del documento y modificarlo	72
6.1.6	Operaciones de edición	73
6.1.7	Buscar y reemplazar	74
6.1.8	Operaciones en filas de tablas	75
6.1.9	Teclas de acceso rápido	76
6.1.10	Botones de estado del texto	77
6.1.11	Ayudantes de entrada	78
6.1.12	Paquetes	79
–	Trabajar con paquetes.....	80
–	Paquetes del corrector ortográfico.....	83
6.1.13	Interfaz XMLData	86
6.1.14	DOM y XMLData	90
6.1.15	Scripting de Authentic	94
6.2	.Objetos	96
6.2.1	Authentic	97
–	Authentic.ApplyTextState.....	100
–	Authentic.attachCallBack.....	101
–	AuthenticView	103
–	Authentic.AutoHideUnusedCommandGroups.....	104
–	Authentic.BaseURL.....	105
–	Authentic.ClearSelection.....	106
–	Authentic.ClearUndoRedo.....	107
–	Authentic.ControlInitialized.....	108
–	Authentic.CreateChild.....	109
–	Authentic.CurrentSelection.....	110
–	Authentic.DesignDataLoadObject.....	111

– Authentic.EditClear.....	112
– Authentic.EditCopy.....	113
– Authentic.EditCut	114
– Authentic.EditPaste.....	115
– Authentic.EditRedo.....	116
– Authentic.EditSelectAll.....	117
– Authentic.EditUndo.....	118
– Authentic.EnableModifications.....	119
– Authentic.EntryHelperAlignment.....	120
– Authentic.EntryHelpersEnabled.....	121
– Authentic.EntryHelperSize.....	122
– Authentic.EntryHelperWindows.....	123
– Authentic.event	124
– Authentic.FindDialog.....	125
– Authentic.FindNext	126
– Authentic.GetAllAttributes.....	127
– Authentic.GetAllowedElements.....	129
– Authentic.GetFileVersion.....	131
– Authentic.GetNextVisible.....	132
– Authentic.GetPreviousVisible.....	133
– Authentic.IsEditClearEnabled.....	134
– Authentic.IsEditCopyEnabled.....	135
– Authentic.IsEditCutEnabled.....	136
– Authentic.IsEditPasteEnabled.....	137
– Authentic.IsEditRedoEnabled.....	138
– Authentic.IsEditUndoEnabled.....	139
– Authentic.IsFindNextEnabled.....	140
– Authentic.IsRowAppendEnabled.....	141
– Authentic.IsRowDeleteEnabled.....	142
– Authentic.IsRowDuplicateEnabled.....	143
– Authentic.IsRowInsertEnabled.....	144
– Authentic.IsRowMoveDownEnabled.....	145
– Authentic.IsRowMoveUpEnabled.....	146
– Authentic.IsTextStateApplied.....	147
– Authentic.IsTextStateEnabled.....	148
– Authentic.LoadXML.....	149
– Authentic.MarkUpView.....	150
– Authentic.Modified	151
– Authentic.Print	152
– Authentic.PrintPreview.....	153
– Authentic.RedrawEntryHelpers.....	154
– Authentic.ReloadToolbars.....	155
– Authentic.ReplaceDialog.....	156
– Authentic.Reset	157
– Authentic.RowAppend.....	158

– Authentic.RowDelete.....	159
– Authentic.RowDuplicate.....	160
– Authentic.RowInsert.....	161
– Authentic.RowMoveDown.....	162
– Authentic.RowMoveUp.....	163
– Authentic.Save	164
– Authentic.SaveButtonAutoEnable.....	165
– Authentic.SavePOST.....	166
– Authentic.SaveXML.....	167
– Authentic.SchemaLoadObject.....	168
– Authentic.SelectionChanged.....	169
– Authentic.SelectionMoveTabOrder.....	170
– Authentic.SelectionSet.....	171
– Authentic.SetUnmodified.....	172
– Authentic.StartEditing.....	173
– Authentic.StartSpellChecking.....	174
– Authentic.TextStateBmpURL.....	175
– Authentic.TextStateToolbarLine.....	176
– Authentic.ToolbarRows.....	177
– Authentic.ToolbarsEnabled.....	178
– Authentic.ToolbarToolipsEnabled.....	179
– Authentic.UICommands.....	180
– Authentic.ValidateDocument.....	181
– Authentic.validationBadData.....	182
– Authentic.validationMessage.....	183
– Authentic.XMLDataLoadObject.....	184
– Authentic.XMLDataSaveUrl.....	185
– Authentic.XMLRoot.....	186
– Authentic.XMLTable.....	187
6.2.2 AuthenticCommand	188
– AuthenticCommand.CommandID.....	189
– AuthenticCommand.Group.....	190
– AuthenticCommand.ShortDescription.....	191
– AuthenticCommand.Name.....	192
6.2.3 AuthenticCommands	193
– AuthenticCommands.Count.....	194
– AuthenticCommands.Item.....	195
6.2.4 AuthenticContextMenu	196
– CountItems	197
– DeleteItem	198
– GetItemText	199
– InsertItem	200
– SetItemText	201
6.2.5 AuthenticDataTransfer	202
– AuthenticDataTransfer.dropEffect.....	203

– AuthenticDataTransfer.getData.....	204
– AuthenticDataTransfer.ownDrag.....	205
– AuthenticDataTransfer.type.....	206
6.2.6 AuthenticEvent	207
– AuthenticEvent.altKey.....	208
– AuthenticEvent.altLeft.....	209
– AuthenticEvent.button.....	210
– AuthenticEvent.cancelBubble.....	211
– AuthenticEvent.clientX.....	212
– AuthenticEvent.clientY.....	213
– AuthenticEvent.ctrlKey.....	214
– AuthenticEvent.ctrlLeft.....	215
– AuthenticEvent.dataTransfer.....	216
– AuthenticEvent.fromElement.....	217
– AuthenticEvent.keyCode.....	218
– AuthenticEvent.propertyName.....	219
– AuthenticEvent.repeat.....	220
– AuthenticEvent.returnValue.....	221
– AuthenticEvent.shiftKey.....	222
– AuthenticEvent.shiftLeft.....	223
– AuthenticEvent.srcElement.....	224
– AuthenticEvent.type.....	225
6.2.7 AuthenticEventContext	226
– EvaluateXPath	227
– GetEventContextType.....	228
– GetNormalizedTextValue.....	229
– GetVariableValue	230
– GetXMLNode	231
– IsAvailable	232
– SetVariableValue	233
6.2.8 AuthenticLoadObject	234
– AuthenticLoadObject.String.....	235
– AuthenticLoadObject.URL.....	236
6.2.9 AuthenticRange	237
– AuthenticRange.AppendRow.....	239
– AuthenticRange.Application.....	240
– AuthenticRange.CanPerformAction.....	241
– AuthenticRange.CanPerformActionWith.....	242
– AuthenticRange.Clone.....	243
– AuthenticRange.CollapsToBegin.....	244
– AuthenticRange.CollapsToEnd.....	245
– AuthenticRange.Copy.....	246
– AuthenticRange.Cut.....	247
– AuthenticRange.Delete.....	248
– AuthenticRange.DeleteRow.....	249

– AuthenticRange.DuplicateRow.....	250
– AuthenticRange.EvaluateXPath.....	251
– AuthenticRange.ExpandTo.....	252
– AuthenticRange.FirstTextPosition.....	253
– AuthenticRange.FirstXMLData.....	254
– AuthenticRange.FirstXMLDataOffset.....	255
– AuthenticRange.GetElementAttributeNames.....	257
– AuthenticRange.GetElementAttributeValue.....	258
– AuthenticRange.GetElementHierarchy.....	259
– AuthenticRange.GetEntityNames.....	260
– AuthenticRange.GetVariableValue.....	261
– AuthenticRange.Goto.....	262
– AuthenticRange.GotoNext.....	263
– AuthenticRange.GotoNextCursorPosition.....	264
– AuthenticRange.GotoPrevious.....	265
– AuthenticRange.GotoPreviousCursorPosition.....	266
– AuthenticRange.HasElementAttribute.....	267
– AuthenticRange.InsertEntity.....	268
– AuthenticRange.InsertRow.....	269
– AuthenticRange.IsCopyEnabled.....	270
– AuthenticRange.IsCutEnabled.....	271
– AuthenticRange.IsDeleteEnabled.....	272
– AuthenticRange.IsEmpty.....	273
– AuthenticRange isEqual.....	274
– AuthenticRange.IsFirstRow.....	275
– AuthenticRange.IsInDynamicTable.....	276
– AuthenticRange.IsLastRow.....	277
– AuthenticRange.IsPasteEnabled.....	278
– AuthenticRange.isSelected.....	279
– AuthenticRange.IsTextStateApplied.....	280
– AuthenticRange.LastTextPosition.....	281
– AuthenticRange.LastXMLData.....	282
– AuthenticRange.LastXMLDataOffset.....	283
– AuthenticRange.MoveBegin.....	285
– AuthenticRange.MoveEnd.....	286
– AuthenticRange.MoveRowDown.....	287
– AuthenticRange.MoveRowUp.....	288
– AuthenticRange.Parent.....	289
– AuthenticRange.Paste.....	290
– AuthenticRange.PerformAction.....	291
– AuthenticRange.Select.....	292
– AuthenticRange.SelectNext.....	293
– AuthenticRange.SelectPrevious.....	294
– AuthenticRange.SetElementAttributeValue.....	295
– AuthenticRange.SetFromRange.....	297

–	AuthenticRange.SetValue.....	298
–	AuthenticRange.Text.....	299
6.2.10	AuthenticSelection	300
–	AuthenticSelection.End.....	301
–	AuthenticSelection.EndTextPosition.....	302
–	AuthenticSelection.Start.....	303
–	AuthenticSelection.StartTextPosition.....	304
6.2.11	AuthenticToolbarButton	305
–	AuthenticToolbarButton.CommandID.....	306
6.2.12	AuthenticToolbarButtons	307
–	AuthenticToolbarButtons.Count.....	308
–	AuthenticToolbarButtons.Item.....	309
–	AuthenticToolbarButtons.NewButton.....	310
–	AuthenticToolbarButtons.NewCustomButton.....	311
–	AuthenticToolbarButtons.NewSeparator.....	312
–	AuthenticToolbarButtons.Remove.....	313
6.2.13	AuthenticToolbarRow	314
–	AuthenticToolbarRowAlignment.....	315
–	AuthenticToolbarRowButtons.....	316
6.2.14	AuthenticToolbarRows	317
–	AuthenticToolbarRows.Count.....	318
–	AuthenticToolbarRows.Item.....	319
–	AuthenticToolbarRows.RemoveRow.....	320
–	AuthenticToolbarRows.NewReader.....	321
6.2.15	AuthenticView	322
–	Eventos	323
–	OnBeforeCopy.....	323
–	OnBeforeCut.....	323
–	OnBeforeDelete.....	324
–	OnBeforeDrop.....	324
–	OnBeforePaste.....	325
–	OnBeforeSave.....	326
–	OnDragOver	326
–	OnKeyboardEvent.....	327
–	OnLoad	329
–	OnMouseEvent.....	329
–	OnSelectionChanged.....	331
–	OnToolBarButtonClicked.....	331
–	OnToolBarButtonExecuted.....	333
–	OnUserAddedXMLNode.....	333
–	AuthenticView.Application.....	334
–	AuthenticView.AsXMLString.....	335
–	AuthenticView.ContextMenu.....	336
–	AuthenticView.CreateXMLNode.....	337

– AuthenticView.DisableAttributeEntryHelper.....	338
– AuthenticView.DisableElementEntryHelper.....	339
– AuthenticView.DisableEntityEntryHelper.....	340
– AuthenticView.DocumentBegin.....	341
– AuthenticView.DocumentEnd.....	342
– AuthenticView.DoNotPerformStandardAction.....	343
– AuthenticView.EvaluateXPath.....	344
– AuthenticView.Event.....	345
– AuthenticView.EventContext.....	346
– AuthenticView.GetToolbarButtonState.....	347
– AuthenticView.Goto.....	348
– AuthenticView.IsRedoEnabled.....	349
– AuthenticView.IsUndoEnabled.....	350
– AuthenticView.MarkupVisibility.....	351
– AuthenticView.Parent.....	352
– AuthenticView.Print.....	353
– AuthenticView.Redo.....	354
– AuthenticView.Selection.....	355
– AuthenticView.SetToolbarButtonState.....	356
– AuthenticView.Undo.....	357
– AuthenticView.UpdateXMLInstanceEntities.....	358
– AuthenticView.WholeDocument.....	359
– AuthenticView.XMLDataRoot.....	360
6.2.16 AuthenticXMLTableCommands	361
– AuthenticXMLTableCommands.AlignHorizontalCenter.....	363
– AuthenticXMLTableCommands.AlignHorizontalJustify.....	364
– AuthenticXMLTableCommands.AlignHorizontalLeft.....	365
– AuthenticXMLTableCommands.AlignHorizontalRight.....	366
– AuthenticXMLTableCommands.AlignVerticalBottom.....	367
– AuthenticXMLTableCommands.AlignVerticalCenter.....	368
– AuthenticXMLTableCommands.AlignVerticalTop.....	369
– AuthenticXMLTableCommands.AppendCol.....	370
– AuthenticXMLTableCommands.AppendRow.....	371
– AuthenticXMLTableCommands.Delete.....	372
– AuthenticXMLTableCommands.DeleteCol.....	373
– AuthenticXMLTableCommands.DeleteRow.....	374
– AuthenticXMLTableCommands.EditProperties.....	375
– AuthenticXMLTableCommands.Insert.....	376
– AuthenticXMLTableCommands.InsertCol.....	377
– AuthenticXMLTableCommands.InsertRow.....	378
– AuthenticXMLTableCommands.JoinDown.....	379
– AuthenticXMLTableCommands.JoinLeft.....	380
– AuthenticXMLTableCommands.JoinRight.....	381
– AuthenticXMLTableCommands.JoinUp.....	382
– AuthenticXMLTableCommands.MayAlignHorizontal.....	383

– AuthenticXMLTableCommands.MayAlignVertical.....	384
– AuthenticXMLTableCommands.MayAppendCol.....	385
– AuthenticXMLTableCommands.MayAppendRow.....	386
– AuthenticXMLTableCommands.MayDelete.....	387
– AuthenticXMLTableCommands.MayDeleteCol.....	388
– AuthenticXMLTableCommands.MayDeleteRow.....	389
– AuthenticXMLTableCommands.MayEditProperties.....	390
– AuthenticXMLTableCommands.MayInsert.....	391
– AuthenticXMLTableCommands.MayInsertCol.....	392
– AuthenticXMLTableCommands.MayInsertRow.....	393
– AuthenticXMLTableCommands.MayJoinDown.....	394
– AuthenticXMLTableCommands.MayJoinLeft.....	395
– AuthenticXMLTableCommands.MayJoinRight.....	396
– AuthenticXMLTableCommands.MayJoinUp.....	397
– AuthenticXMLTableCommands.MaySplitHorizontal.....	398
– AuthenticXMLTableCommands.MaySplitVertical.....	399
– AuthenticXMLTableCommands.SplitHorizontal.....	400
– AuthenticXMLTableCommands.SplitVertical.....	401
6.2.17 XMLData	402
– XMLData.AppendChild.....	403
– XMLData.CountChildren.....	404
– XMLData.CountChildrenKind.....	405
– XMLData.EraseAllChildren.....	406
– XMLData.EraseChild.....	407
– XMLData.EraseCurrentChild.....	408
– XMLData.GetChild	409
– XMLData.GetChildAttribute	410
– XMLData.GetChildElement	411
– XMLData.GetChildKind	412
– XMLData.GetCurrentChild	413
– XMLData.GetFirstChild	414
– XMLData.GetNamespacePrefixForURI	415
– XMLData.GetNextChild	416
– XMLData.GetTextValueXMLDecoded	417
– XMLData.HasChildren	418
– XMLData.HasChildrenKind	419
– XMLData.InsertChild	420
– XMLData.InsertChildAfter	421
– XMLData.InsertChildBefore	422
– XMLData.IsSameNode	423
– XMLData.Kind	424
– XMLData.MayHaveChildren	425
– XMLData.Name	426
– XMLData.Parent	427
– XMLData.SetTextValueXMLDecoded	428

– XMLData.TextValue.....	429
6.3 ...Enumeraciones	430
6.3.1 SPYAuthenticActions	431
6.3.2 SPYAuthenticCommand	432
6.3.3 SPYAuthenticCommandGroup	434
6.3.4 SPYAuthenticDocumentPosition	435
6.3.5 SPYAuthenticElementActions	436
6.3.6 SPYAuthenticElementKind	437
6.3.7 SPYAuthenticEntryHelperWindows	438
6.3.8 SPYAuthenticMarkupVisibility	439
6.3.9 SPYAuthenticToolbarAllignment	440
6.3.10 SPYAuthenticToolbarButtonState	441
6.3.11 SPYXMLDataKind	442

7 Aplicaciones web ASP.NET **444**

8 Información sobre licencias **446**

8.1 ...Distribución electrónica de software	447
8.2 ...Derechos de propiedad intelectual	448
8.3 ...Altova Authentic: contrato de licencia para el usuario final	449

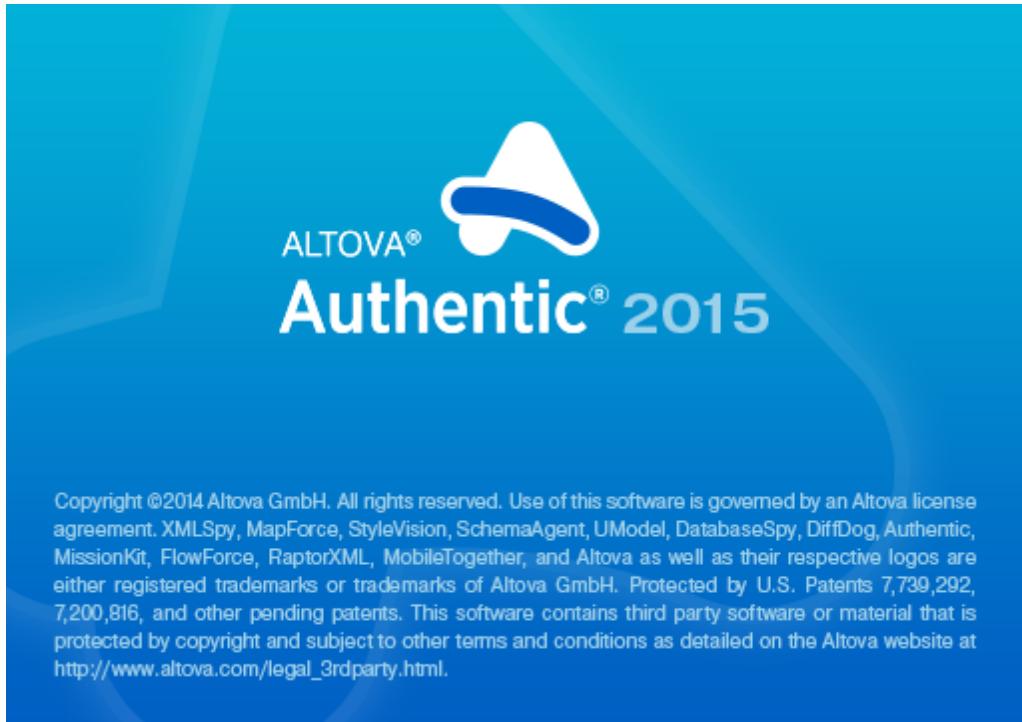
Índice

Altova Authentic 2015 Browser Edition

Altova Authentic Browser Edition

Altova Authentic Browser Edition

Altova® Authentic® 2015 Browser Edition permite crear y editar contenido XML y de BD desde una interfaz de usuario muy parecida a la de un procesador de texto. La edición Browser Edition se puede incrustar en cualquier página web para que el usuario edite el contenido desde el explorador web. La edición Browser Edition está disponible en forma de complemento para los exploradores **Microsoft Internet Explorer** y **Mozilla Firefox**.



Hay dos versiones de Authentic Browser: (i) la versión **Enterprise Edition** (para la que es necesario comprar un código clave) ofrece funciones avanzadas como los [scripts de Authentic](#)) y (ii) la versión **Community Edition**, que es gratuita. Además Altova ofrece [una versión segura y una versión no segura](#) de cada edición de Authentic Browser.

Altova Authentic 2015 Browser Edition

Información sobre este manual

Información sobre este manual

Ese manual del usuario de Authentic Browser se divide en varias secciones:

- La [introducción](#) explica (i) las [ventajas de usar Authentic Browser](#), (ii) [su funcionamiento](#) y (iii) las diferencias entre las [versiones de Authentic Browser](#).
- La sección [Configuración del servidor](#) describe los pasos necesarios para configurar un servidor para un proyecto de Authentic Browser e incluye una descripción detallada de la [página HTML para Authentic Browser](#) y una sección que explica cómo usar los paquetes de extensión de Authentic Browser para [instalaciones por encargo](#).
- La sección [Configuración del cliente](#) explica cómo [instalar el complemento Authentic Browser en los exploradores cliente](#).
- La referencia del usuario se divide en tres grandes secciones ([Mecanismos](#), [Objetos](#) y [Enumeraciones](#)) que describen los mecanismos, los objetos y las enumeraciones que se utilizan para crear y personalizar la vista Authentic en Authentic Browser.
- La sección [Aplicaciones web ASP .NET](#) describe el control de servidor web ASP .NET que Altova ofrece para facilitar la implementación del complemento Authentic Browser que se integra completamente con Visual Studio .NET.

Documentación relacionada

También puede leer la documentación de Altova sobre estos productos y componentes relacionados:

- La documentación para crear archivos SPS (StyleVision Power Stylesheet) en Altova StyleVision disponible en el [sitio web de Altova](#). Los archivos SPS controlan la vista Authentic de un documento XML. Consulte esta documentación para desarrollar la interfaz de la vista Authentic para editar datos XML.
- La documentación de la vista Authentic. Consulte el tutorial del manual de Authentic Desktop disponible en el [sitio web de Altova](#).

Altova Authentic 2015 Browser Edition

Introducción

Introducción

Altova® Authentic® 2015 Browser Edition permite editar documentos XML a partir de archivos SPS (**StyleVision Power Stylesheet**) creados con **Altova StyleVision**. Se trata de una solución única que permite editar contenido XML en vivo desde el explorador web de cualquier equipo de escritorio (cliente) de la empresa.

Esta sección abarca estos tres temas:

- [Ventajas de la edición Authentic Browser](#): explica el uso que puede dar a Authentic Browser y sus ventajas.
- [Funcionamiento](#): explica brevemente el funcionamiento de Authentic Browser dependiendo del sistema utilizado.
- [Versiones de la edición Authentic Browser](#): enumera y explica todas las versiones disponibles de Authentic Browser.

1 Ventajas de la edición Authentic Browser

Estas son las principales ventajas del complemento Authentic Browser para edición XML en exploradores web:

- Permite a varios acceder a documentos XML y editarlos desde el explorador web. Es compatible con **Microsoft Internet Explorer 5.5 (o superior)** y **Mozilla Firefox**.
- Facilita la implementación en toda la organización y su mantenimiento. Al mismo tiempo reduce el coste total de propiedad.
- Está basado en estándares abiertos como XML Schema y XSLT.
- Es totalmente compatible con Unicode.
- Utiliza la vista Authentic de Altova, que se basa en el explorador Internet Explorer. La vista Authentic sirve para editar archivos XML en una interfaz WYSIWYG (es decir, el usuario no ve el código XML subyacente).
- No requiere la implementación de otro software porque Authentic Browser es un complemento para exploradores web.
- Authentic Browser es un control ActiveX en el que el objeto [Authentic](#) define la interfaz COM. El modelo de objetos completo se describe en la sección [Referencia del usuario: Objetos](#) de este manual.

2 Funcionamiento

Para implementar un proyecto de Authentic Browser necesitará un equipo servidor y, como mínimo, un equipo cliente conectado al servidor.

Servidor de Authentic Browser

El servidor de Authentic Browser lleva a cabo estas funciones:

- El servidor almacena los archivos relacionados con el documento XML que se va a editar.
Se trata de estos archivos:
 1. El esquema XML (XSD) en el que se basa el documento XML editable.
 2. El documento XML que se va a editar.
 3. El archivo SPS o PXF que controla el diseño y los mecanismos de entrada del documento XML en la vista Authentic.
- El servidor también almacena la [página HTML para el complemento Authentic Browser](#). Esta página HTML es el punto de acceso para la edición en la vista Authentic y contiene instrucciones para acceder al documento XML, además de hacer de contenedor para la ventana de la vista Authentic donde se carga y edita el documento XML. Para acceder a esta página escriba su URL en el explorador cliente.
- Si implementa la edición Authentic Browser Enterprise Edition, la licencia de la edición Enterprise Edition se almacena en el servidor. Altova puede emitir licencias de la edición Enterprise Edition para uno o varios servidores.
- Si tiene pensado instalar el complemento Authentic Browser por encargo, el servidor almacena los paquetes de extensión del complemento (archivos CAB, XPI y CRX) para la descarga e instalación del complemento en equipos cliente. De lo contrario, el complemento se instala directamente en el explorador web cliente.

Las instrucciones para preparar el servidor se describen en la sección [Configuración del servidor](#).

Clientes de Authentic Browser

En la vista Authentic puede ver y editar documentos XML que tengan asignados un archivo SPS o PXF. La edición en la vista Authentic se lleva a cabo en equipos cliente que deben configurarse de la siguiente manera:

- Cada cliente debe tener instalado uno de estos exploradores web: **Internet Explorer 5.5 o superior (32 o 64 bits)** y **Firefox (32 bits)**.
- Además debe instalar **Internet Explorer 5.5 o superior** en cada equipo cliente. El motivo es que la interfaz de la vista Authentic (que se presenta en la ventana del explorador) se genera usando Internet Explorer.
- Si no tiene pensada una instalación por encargo, instale el complemento Authentic Browser en los exploradores cliente directamente.

Las instrucciones para preparar los clientes se describen en la sección [Configuración de clientes](#).

Mecanismos de Authentic Browser

Cuando el servidor y los clientes ya están preparados, el usuario introduce la URL de la página HTML del complemento de Authentic en el explorador cliente. Si el complemento no está instalado todavía en el explorador cliente, la página HTML puede incluir instrucciones para realizar una instalación por encargo.

Cuando el complemento ya está instalado en el cliente, el código de la página HTML abre una ventana de edición de la vista Authentic dentro de la ventana del explorador. El documento XML que desea editar se carga en la ventana de la vista Authentic desde el servidor y el usuario puede empezar a editarlo y a guardar cambios en el documento XML directamente.

3 Versiones de la edición Authentic Browser

Altova ofrece estas versiones del complemento Authentic Browser para exploradores web:

- *Idioma*: Altova ofrece versiones en español (ES), inglés (EN), alemán (DE) y japonés (JA).
- *Seguridad*: Altova ofrece versiones seguras (*Trusted*) y no seguras (*Untrusted*).
- *Explorador*: Altova ofrece complementos para Microsoft Internet Explorer (32 y 64 bits) y Mozilla Firefox (32 bits).

Por cada complemento (para Microsoft Internet Explorer y Firefox) hay una versión segura y una versión no segura en todos los idiomas disponibles (español, inglés, alemán y japonés). En cada equipo cliente puede instalar todas las versiones que quiera del complemento Authentic Browser. Cada versión aparece por separado en el administrador de complementos del explorador web.

A continuación puede ver las versiones en español, inglés, alemán y japonés con sus Classid (para archivos CAB) y sus tipos MIME (para archivos XPI y CRX). El Classid o tipo MIME correspondiente debe especificarse en la [página HTML para el complemento de Authentic](#).

- **Archivos CAB y sus Classid (en Internet Explorer 32/64 bits se usa el mismo ID de Classid)**

ES	<i>Trusted</i>	23B503E7-269B-45CE-BAB2-22AA97BED8E2
ES	<i>Untrusted</i>	8AD3EF86-AC1E-4574-8C13-DE5B6CBECEBE
EN	<i>Trusted</i>	B4628728-E3F0-44a2-BEC8-F838555AE780
EN	<i>Untrusted</i>	A5985EA9-3332-4ddf-AD7F-F6E98BFEAF94
DE	<i>Trusted</i>	91DDF44A-DFD1-4F47-8EE3-4CBE874584F7
DE	<i>Untrusted</i>	28A640E8-EAEE-4B5D-BEBE-BFA956081E66
JA	<i>Trusted</i>	5B15DB5A-1720-4264-BB65-70C3F7A860DA
JA	<i>Untrusted</i>	4B9512D2-A3D3-46e3-82C1-34248BBDC58

- **Tipos MIME para archivos XPI y CRX**

ES	<i>Trusted</i>	application/x-authentic-scriptable-plugin-spanish
ES	<i>Untrusted</i>	application/x-authentic-scriptable-plugin-untrusted-spanish
EN	<i>Trusted</i>	application/x-authentic-scriptable-plugin
EN	<i>Untrusted</i>	application/x-authentic-scriptable-plugin-untrusted
DE	<i>Trusted</i>	application/x-authentic-scriptable-plugin-german
DE	<i>Untrusted</i>	application/x-authentic-scriptable-plugin-untrusted-german

JA	<i>Trusted</i>	application/x-authentic-scriptable-plugin-japanese
JA	<i>Untrusted</i>	application/x-authentic-scriptable-plugin-untrusted-japanese

Todas estas versiones se pueden descargar del [sitio web de Altova](#).

Debe tener en cuenta algunos aspectos de las versiones del complemento Authentic Browser:

- Todas las versiones son Unicode y son totalmente compatibles con múltiples juegos de caracteres del documento XML. Las versiones Unicode exigen el uso de Windows 8, Windows 7, Windows Vista o Windows XP en el equipo cliente.
- También hay un archivo CAB distinto para cada versión Internet Explorer de 32 y 64 bits. Los Classid de los dos archivos CAB (para exploradores IE de 32 y 64 bits) son idénticos, es decir, los que aparecen en la tabla anterior para las versiones ES/EN/DE/JA seguras y no seguras.
- La versión segura **Trusted** no permite acceder a archivos locales y, por tanto, se considera segura para secuencias de comandos. Se puede usar en casos de uso basados en exploradores web y se puede invocar desde cualquier página web sin causar alertas de seguridad en el lado cliente.
- La versión no segura Untrusted está diseñada para la implementación en intranets o para usar el complemento Authentic Browser como control ActiveX en su aplicación. Ofrece acceso a archivos locales y, por tanto, no se considera segura para secuencias de comandos. Si intenta usar esta versión desde una ventana del explorador, aparecerá un aviso. Puede elegir si los controles ActiveX se deben habilitar o deshabilitar o si el explorador debe pedir permiso para habilitarlos (consulte el apartado [Internet Explorer 9](#) para obtener más información).

Nota: para instalar y configurar el servicio de explorador (p. ej. Internet Information Services) consulte la documentación del proveedor.

Altova Authentic 2015 Browser Edition

Configuración del servidor

Configuración del servidor

Varios pasos son necesarios para configurar el servidor para el complemento Authentic Browser.

Configurar el servicio de explorador

Instale y [configure el servicio de explorador de su servidor](#). Si usa Internet Information Services (IIS) de Microsoft, tenga en cuenta que durante la instalación se crea el directorio predeterminado Inetpub, que contiene varias subcarpetas. El directorio raíz del servidor sería entonces // Inetpub/wwwroot. Este es el directorio al que se llega con la dirección IP del servidor, a no ser que indique otro directorio como directorio raíz (en la configuración del servicio de explorador). Para más información sobre cómo instalar y configurar el servicio de explorador, consulte la documentación del proveedor.

Configurar los archivos XSD, XML y SPS/PXF

Los archivos SPS permiten representar documentos XML en un formato editable en la vista Authentic. Están basados en esquemas XML (archivos XSD) y se diseñan con [Altova StyleVision](#). El archivo SPS, junto con el archivo XSD y el documento XML que se debe editar, debe almacenarse en una ubicación de la red a la que tengan acceso todos los equipos clientes (por lo general, se almacenan en el servidor de Authentic Browser). El apartado [Archivos XSD, XML y SPS/PXF](#) describe cómo preparar el servidor para esto.

Crear la página HTML para el complemento Authentic Browser

La [página HTML para el complemento Authentic Browser](#) es el punto de acceso para editar en la vista Authentic. Contiene instrucciones para acceder al documento XML y sirve de contenedor de la ventana de la vista Authentic en la que se carga y edita el documento XML. Para acceder a esta página se debe escribir su URL en los exploradores clientes. Esta página HTML debe crearse correctamente y debe almacenarse en el servidor. Para más información consulte el apartado [Página HTML para el complemento de Authentic](#).

Almacenar paquetes de extensión para instalación por encargo

Descargue el complemento Authentic Browser (un archivo comprimido CAB, XPI o CRX) del sitio web de Altova y guárdelo en cualquier ubicación del servidor. Si implementa la edición Authentic Browser Enterprise Edition, el paquete debe almacenarse en el servidor donde está registrada la licencia para la edición Enterprise Edition. **Nota importante:** no desempaque este archivo.

Por cada complemento hay una versión segura y una versión no segura en todos los idiomas disponibles (español, inglés, alemán y japonés) en cuatro formatos diferentes (CAB de 32 y 64 bits, XPI y CRX). Para más información sobre los [formatos de archivo](#) y las diferentes [versiones](#), consulte los apartados de esta sección.

Nota: antes de instalar el complemento Authentic Browser, compruebe que no hay ninguna versión antigua de Authentic Browser en ejecución. De lo contrario, la versión más reciente no se registrará correctamente y la instalación será defectuosa. Si esto ocurre, registro el complemento ejecutando `regsvr32 C:\Windows\Downloaded Program Files\AuthenticPlugin.dll`. (Recuerde que se necesitan privilegios de administrador para ejecutar el programa `regsvr32.exe`.)

1 IIS: configurar el servicio de explorador

Internet Information Services (IIS) 6 de Microsoft solamente sirve los tipos de archivo que se definen en los tipos MIME del sitio (sitio web o carpeta). Por tanto, los tipos de archivo necesarios deben añadirse a la lista de tipos MIME del sitio.

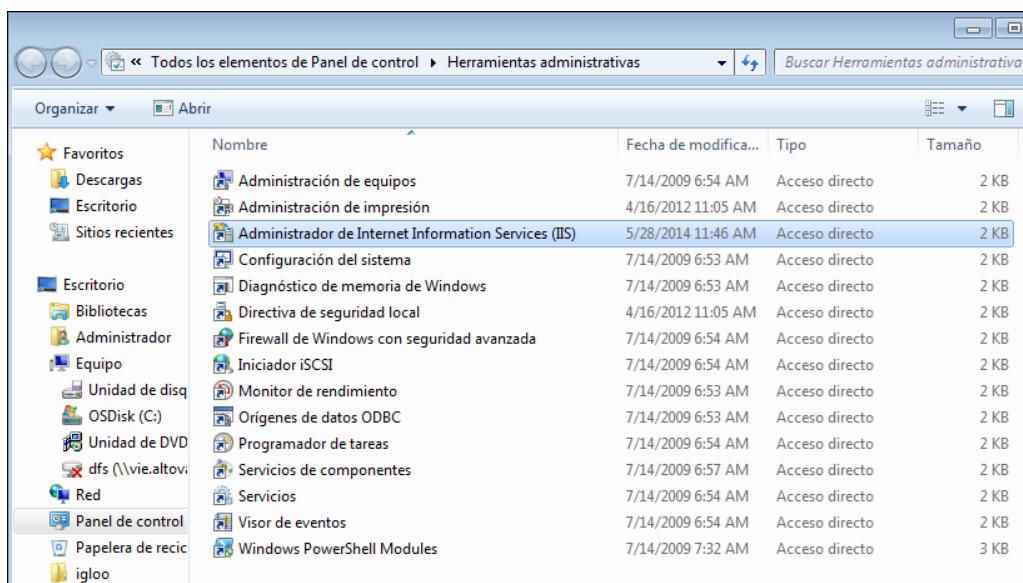
Para trabajar con el complemento Authentic Browser son necesarios estos tipos de archivo:

Extensión	Tipo MIME	Comentario
xsd	text/plain	
sps	text/plain	
pxf	application/x-zip-compressed	
xpi	application/x-xpinstall	Para Firefox

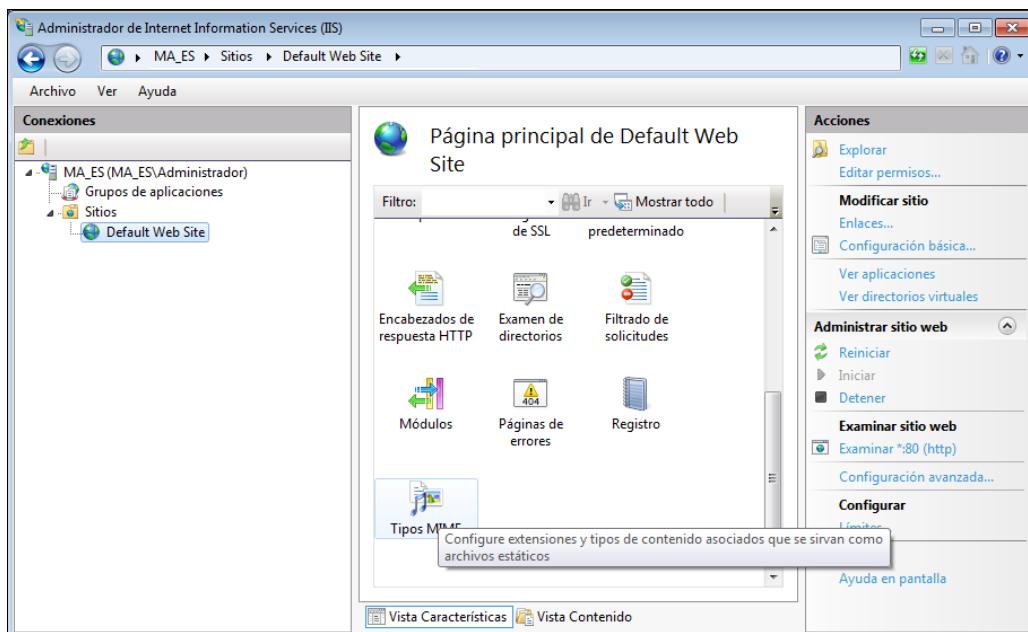
Agregar tipos MIME para un sitio en Internet Information Services

A continuación explicamos cómo agregar un tipo MIME a la lista de tipos MIME de un sitio concreto en un equipo Windows 7. El proceso es similar en los demás sistemas compatibles (Windows XP, Windows Vista y Windows 8).

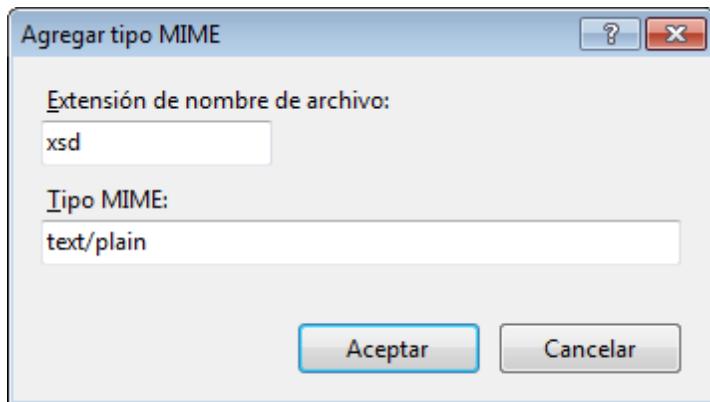
1. Abra el panel de control y haga doble clic en **Herramientas administrativas**.
2. En el cuadro de diálogo que aparece (*imagen siguiente*) haga doble clic en **Administrador de Internet Information Services (IIS)**.



3. Aparece el cuadro de diálogo "Internet Information Services" (*imagen siguiente*). Seleccione el sitio pertinente (sitio web o carpeta) en el panel izquierdo y en el panel central haga doble clic **Tipos MIME**.



4. Esto abre el cuadro de diálogo "Agregar tipo MIME" (*imagen siguiente*). Escriba la extensión de archivo correspondiente y su tipo MIME (consulte la tabla que aparece al principio del apartado para encontrar el tipo MIME y la extensión correcta).



5. Haga clic en **Aceptar** para confirmar.

Para más información sobre cómo configurar el acceso remoto a una carpeta en un servidor interno a través de WebDAV, consulte [este artículo](#).

2 Archivos XSD, XML y SPS/PXF

El principal objetivo de implementar un proyecto de Authentic Browser es ver y editar documentos XML en la vista Authentic. Este formato oculta el marcado XML del documento XML y habilita la edición del documento en una interfaz WYSIWYG con un diseño personalizado. El diseño y los mecanismos de entrada de datos de un documento XML en la vista Authentic se definen en un archivo SPS. Por tanto, el archivo SPS, el archivo XML y el esquema XML (archivo XSD) en el que se basan el archivo SPS y los documentos XML deben almacenarse en la red para que los clientes de Authentic Browser puedan acceder a ellos.

Archivos SPS

Los archivos SPS se crean con [Altova StyleVision](#). Se basa en el mismo esquema XML que el archivo XML. En StyleVision el diseñador del archivo SPS puede usar funciones de diseño gráfico para diseñar el SPS, que determina el diseño y los mecanismos de entrada de datos del documento XML cuando se abre en la vista Authentic.

En la [página HTML para el complemento Authentic Browser](#) se especifica la ubicación del archivo XML, del archivo SPS y del archivo XSD. El archivo XML se carga en la ventana de la vista Authentic (en la página HTML) y se presenta siguiendo el diseño especificado en el archivo SPS. Tanto el archivo XML como el archivo SPS están basados en el mismo esquema.

Nota: para garantizar el correcto funcionamiento de Authentic Browser, **el esquema debe estar asignado como esquema principal** en el archivo SPS. Para obtener más información sobre cómo crear un archivo SPS (StyleVision Power Stylesheet), consulte el [manual del usuario de Altova StyleVision](#). También puede obtener más información sobre Altova StyleVision en el [sitio web de Altova](#)

Archivos XML

El usuario de la vista Authentic edita el documento XML en un equipo cliente a través de la [página HTML para el complemento de Authentic](#). Esta página contiene una ventana de la vista Authentic donde se carga el archivo XML. El usuario de la vista Authentic edita el documento en el explorador cliente y guarda los cambios en el documento XML.

Archivos XSD

Los archivos XSD tienen dos funciones. Sirven para validar el documento XML y aporta la estructura genérica de documento en el que se basa el SPS. Tanto el archivo XML como el archivo SPS hacen referencia al archivo XSD. Por tanto, el archivo XSD también se debe almacenar en la ubicación correcta, tal y como se indica en los archivos XML y SPS.

Ubicación de los archivos XSD, XML y SPS/PXF

Los archivos XSD, XML y SPS (o PXF) deben guardarse en ubicaciones de la red a las que puedan acceder todos los equipos clientes. La mejor ubicación es el servidor de Authentic Browser. Lo mejor es guardar estos archivos en una sola carpeta del servidor y especificar en ellos referencias a los demás en forma de rutas de acceso relativas. Por ejemplo, el archivo XML se puede crear con la referencia al XSD en forma de ruta de acceso relativa.

Nota sobre archivos SPS basados en BD

Si quiere usar el complemento Authentic Browser para ver o editar bases de datos (BD) mediante un archivo SPS basado en una BD, deberá configurar el cliente para que pueda conectarse a la BD correctamente.

Información sobre la conexión en el archivo SPS: toda la información necesaria para conectarse a la BD se almacena en una cadena de conexión en el archivo SPS. Esta cadena se crea con StyleVision cuando se crea el archivo SPS. El mecanismo utilizado para acceder a bases de datos MS Access es distinto al de las demás bases de datos, que utilizan conexiones ADO. Esta es la configuración que debe utilizar para cada tipo de mecanismo de conexión:

- **Para bases de datos MS Access:** en la cadena de conexión debe usar una ruta de acceso UNC para que los clientes se puedan conectar correctamente con a BD. Esta ruta de acceso UNC se especifica cuando la SPS se crea con StyleVision. Además, compruebe que la carpeta que contiene la BD (o alguna carpeta antecesora) está compartida. En este equipo también debe habilitar el uso compartido avanzado. En los clientes no es necesario cambiar la configuración.
- **Nota:** el formato de la ruta de acceso UNC que se utiliza en la cadena de conexión es \nombreServidor\carpetaCompartida\ruta\archivo.mdb.
- **Para conexiones ADO:** la cadena de conexión ADO del archivo SPS se especifica al crear el SPS con StyleVision. Contiene toda la información necesaria para la conexión, incluida la información de seguridad. Compruebe que el controlador utilizado para probar la cadena de conexión en StyleVision también está instalado en todos los equipos cliente que hospedarán el complemento Authentic Browser. Esto permite al archivo SPS conectarse correctamente a la BD desde los clientes. Recuerde también que las conexiones ADO solamente funcionan con rutas de acceso locales y no con direcciones URL de tipo `http://`.

Nota sobre archivos PXF

Un diseño SPS que utiliza XSLT 2.0 se puede guardar en el formato de archivo PXF (Power XML Form). Este formato de archivo fue desarrollado por Altova para empaquetar el diseño SPS con los archivos relacionados (como el archivo de esquema, el archivo XML de origen, los archivos de imagen utilizados en el diseño y los archivos XSLT necesarios para transformar el XML de origen en el formato de salida). La ventaja de usar este formato de archivo es que todos los archivos necesarios para la edición en la vista Authentic y para generar los resultados de la vista Authentic se pueden distribuir en un solo archivo con total comodidad.

Nota: si el archivo PXF está ubicado en un servidor web y se usará con el complemento Authentic Browser, compruebe que el servidor no bloquea el archivo. Para ello debe añadir este tipo MIME para las extensiones de archivo PXF : `application/x-zip-compressed`.

3 Página HTML para el complemento de Authentic

La página HTML para el complemento de Authentic desempeña estas funciones:

1. La primera vez que se abre la página HTML para el complemento de Authentic en un cliente, el código de la página HTML descarga el complemento Authentic Browser del servidor y lo instala en el cliente. Este código debe estar escrito correctamente para poder identificar el archivo correcto en el servidor. El formato del archivo que se descarga (CAB, XPI o CRX) dependerá del explorador del cliente. El código también puede variar porque Internet Explorer y Firefox no procesan código de la misma manera.
2. El código de la página HTML configura la interfaz de la vista Authentic dentro de la ventana del explorador, incluidas las dimensiones de la interfaz y la ubicación de los archivos XML, XSD y SPS.
3. Especifica qué archivo XML se debe editar y en qué esquema o archivo SPS se basa.
4. Contiene, dentro de elementos HTML `SCRIPT`, definiciones para las subrutinas y el control de eventos. Por ejemplo, puede indicar qué acción se lleva a cabo cuando se hace clic en un botón de la página HTML. Para ver algunos ejemplos consulte los apartados [Internet Explorer. Ejemplo nº1: ejemplo básico](#) y [Firefox. Ejemplo nº1: ejemplo básico](#).

Nota: si implementa la edición **Enterprise Edition** del complemento **Authentic Browser**, la página HTML debe instalarse en el servidor para el que se registró la licencia de la edición Enterprise Edition.

Incrustar el complemento Authentic Browser

Para usar el complemento Authentic Browser es necesario incrustar un objeto que identifique el complemento en la página HTML que descarga el complemento. Esto se hace mediante estos elementos y atributos HTML:

- Internet Explorer: `<OBJECT clsid="clsid:<CLSID>" />`
- Firefox: `<embed type="<MIMETYPE>" />`

Para más información sobre qué valores usar para CLSID y MIMETYPE consulte el apartado [Versiones del complemento Authentic Browser](#).

Resumen de esta sección

Los apartados de esta sección describen cómo implementar las funciones anteriores en la página HTML en cada tipo de explorador web. Esto se debe a que cada explorador descarga el DLL del complemento Authentic Browser de una manera diferente:

- [Licencia para la edición Enterprise Edition](#): este apartado describe el mecanismo para asignar licencias a la edición Enterprise Edition de Authentic Browser.
- [Internet Explorer](#): este apartado describe cómo descargar un archivo CAB mediante un elemento HTML `OBJECT`.
- [Firefox](#): este apartado describe cómo descargar un archivo XPI mediante un elemento HTML `EMBED`.
- [Otros exploradores web](#): describe cómo detectar qué tipo de explorador está realizando la solicitud y cómo descargar la versión correcta del complemento para ese explorador.

En estos apartados encontrará descripciones y ejemplos de uso de los elementos HTML `OBJECT`, `EMBED` y `SCRIPT` y ejemplos de páginas HTML que invocan el complemento Authentic Browser. Si

necesita información sobre algún objeto en particular, consulte la sección [Objectos](#) de la [Referencia](#).

3.1 Licencia para la edición Enterprise Edition

Altova ofrece dos ediciones del complemento Authentic Browser:

1. **Enterprise Edition** es una edición que habilita características avanzadas del archivo SPS. Esta edición requiere una licencia.
2. **Community Edition** es una edición gratuita para la que no se necesita una licencia.

La licencia para la edición Enterprise Edition se puede comprar en la [tienda en línea de Altova](#).

Resumen: cómo configurar la licencia para la edición Enterprise Edition

A continuación resumimos los pasos para configurar correctamente la información sobre licencia para Authentic Browser Enterprise Edition.

- Authentic Browser Enterprise Edition requiere esta información sobre la licencia: (i) **nombre del servidor** para el que se registró la licencia, (ii) **nombre de la compañía** para la que se registró la licencia y (iii) la **clave de licencia**. Esta información está en el correo electrónico que recibirá después de comprar la licencia para Authentic Browser Enterprise Edition.
- La información sobre la licencia debe estar en la [página HTML para el complemento de Authentic](#). Más abajo explicamos cómo hacer esto.
- Si la información sobre la licencia indicada en la [página HTML para el complemento de Authentic](#) es válida, las funciones de Authentic Browser Enterprise Edition se desbloquearán. De lo contrario, solamente estarán disponibles las funciones de la edición Community Edition.
- La [página HTML para el complemento de Authentic](#) debe almacenarse en el servidor para el que se registró la licencia.

Cómo introducir la información sobre la licencia

Los tres parámetros de la licencia (nombre del servidor, nombre de la compañía y clave de licencia) deben insertarse en la [página HTML para el complemento de Authentic](#). Esto puede hacerse de varias maneras:

- Como valores de parámetro del objeto OBJECT o EMBED. Esto se explica en el subapartado correspondiente de los apartados [Internet Explorer](#) y [Firefox](#).
- Si se trata de una página HTML compatible con cualquier tipo de explorador, los parámetros de la licencia se pueden registrar como en el ejemplo que aparece en el apartado [Página HTML para detectar el explorador](#)
- Los parámetros de la licencia también se pueden insertar en el objeto directamente, tal y como muestra el ejemplo siguiente (que es una adaptación del ejemplo del apartado [Página HTML para detectar el explorador](#))

```
<SCRIPT LANGUAGE=javascript>
// event subscription if running on Firefox
if (isFirefoxOnWindows())
{
    objPlugIn.addEventListener("ControlInitialized",
InitAuthenticPluginPage, false);
}
</SCRIPT>
<SCRIPT LANGUAGE="javascript" FOR=objPlugIn EVENT="ControlInitialized">
```

```
// event subscription if running on Internet Explorer
if ( isIEOnWindows() )
{
    InitAuthenticPluginPage();
}
</SCRIPT>

<SCRIPT type="text/javascript" LANGUAGE="javascript" >
function InitAuthenticPluginPage( )
{
    var serverstr='DevAuthBrowTest';
    var basedir='Authentic/';
    objPlugIn.LicServer = 'DevAuthBrowTest';
    objPlugIn.LicCompany = 'Altova';
    objPlugIn.LicKey = 'XXXXXXXXXX';
    objPlugIn.SchemaLoadObject.URL = 'http://' + serverstr + basedir +
'OrgChart.xsd';
    objPlugIn.XMLDataLoadObject.URL = 'http://' + serverstr + basedir +
'OrgChart.xml' ;
    objPlugIn.DesignDataLoadObject.URL = 'http://' + serverstr + basedir +
'OrgChart.sps';
    objPlugIn.StartEditing();
}
</SCRIPT>
```

3.2 Internet Explorer

Si la página HTML para el complemento de Authentic se abre en Internet Explorer (32 o 64 bits), debe incluir estos elementos:

- Un elemento HTML `OBJECT` que (i) provoca la descarga del DLL para el complemento de Authentic (32 o 64 bits) desde el servidor al cliente y (ii) especifica las dimensiones de la ventana de la vista Authentic en el explorador del cliente. El elemento `OBJECT` contiene la ubicación del complemento Authentic Browser. Recuerde que el complemento de Authentic para Internet Explorer puede ser de 32 o de 64 bits, así que debe descargar la versión correcta (archivo .CAB) en el cliente. En el apartado [Ejemplo para cualquier explorador](#) puede ver un ejemplo de código que comprueba automáticamente la versión de Internet Explorer y descarga el complemento de Authentic correspondiente.
- Uno o varios elementos HTML `SCRIPT` para definir las subrutinas y el control de eventos. Un elemento `SCRIPT` se puede usar para especificar qué documento XML se debe editar y el esquema XML y el archivo SPS en el que se basa.

Este apartado tiene varios subapartados:

- [El elemento OBJECT](#): describe cómo se utiliza el elemento HTML `OBJECT` en una página HTML para el complemento de Authentic.
- [El elemento SCRIPT](#): describe cómo se utilizan los elementos `SCRIPT` en una página HTML para el complemento de Authentic.
- Ejemplos de páginas HTML completas: [IE. Ejemplo nº1: ejemplo básico](#) y [IE. Ejemplo nº2: ordenar una tabla](#).

Si necesita información sobre algún objeto en particular consulte el apartado [Objetos](#) de la [Referencia](#).

Nota: el complemento Authentic Browser es compatible con **Internet Explorer versión 5.5 o superior**.

3.2.1 El elemento OBJECT

El elemento OBJECT cumple estas funciones:

- Pone un nombre al complemento Authentic Browser (a través del atributo `id`).
- Selecciona qué [versión del complemento Authentic Browser](#) se debe utilizar (con el valor del atributo `classid`).
- Especifica un archivo CAB y un número de versión (atributo `codebase`). El archivo CAB contiene un DLL que registra un objeto de COM (el complemento Authentic Browser) con un ID que es el valor del atributo `classid`. Por lo general, tanto la versión de 32 bits como la de 64 bits del complemento se almacenan en el servidor. Cada versión se identifica con un nombre de archivo distinto. El atributo `codebase` especifica el nombre del archivo CAB necesario (ver [ejemplo de código más abajo](#)). Para los archivos CAB de 32 y 64 bits se usa el mismo `Classid` (consulte la tabla del apartado [Versiones de la edición Authentic Browser](#)).

Nombre de archivo para la versión de 32 bits: `AuthenticBrowserEdition.CAB`

Nombre de archivo para la versión de 64 bits: `AuthenticBrowserEdition_x64.CAB`

- Especifica la dimensión de la ventana de la vista Authentic en el explorador del cliente (mediante el atributo `style`).
- Puede especificar otros parámetros.

Más abajo puede ver un ejemplo de elemento HTML OBJECT. Este elemento selecciona la versión Trusted Unicode (con el valor dado en el atributo `classid`) y establece unas dimensiones de 600 x 600 píxeles para la ventana de la vista Authentic en el explorador del cliente. Más abajo se describen también todos los atributos y parámetros que se pueden usar en el elemento OBJECT.

Nota: el número de versión dado más abajo no es necesariamente el número de la versión actual. Para más información consulte la descripción del atributo [codebase](#).

Para el complemento Authentic Browser de 32 bits:

```
<OBJECT id="objPlugIn" style="WIDTH:600px; HEIGHT:500px"
        codeBase="http://yourserver/cabfiles/
AuthenticBrowserEdition.CAB#Version=12,3,0,0"
        classid="clsid:B4628728-E3F0-44a2-BEC8-F838555AE780">
<PARAM NAME="XMLDataURL" VALUE="http://yourserver/OrgChart.xml">
<PARAM NAME="SPSDataURL" VALUE="http://yourserver/OrgChart.sps">
<PARAM NAME="SchemaDataURL" VALUE="http://yourserver/OrgChart.xsd">
</OBJECT>
```

Para el complemento Authentic Browser de 64 bits:

```
<OBJECT id="objPlugIn" style="WIDTH:600px; HEIGHT:500px"
        codeBase="http://yourserver/cabfiles/
AuthenticBrowserEdition_x64.CAB#Version=12,3,0,0"
        classid="clsid:B4628728-E3F0-44a2-BEC8-F838555AE780">
<PARAM NAME="XMLDataURL" VALUE="http://yourserver/OrgChart.xml">
<PARAM NAME="SPSDataURL" VALUE="http://yourserver/OrgChart.sps">
<PARAM NAME="SchemaDataURL" VALUE="http://yourserver/OrgChart.xsd">
</OBJECT>
```

id

El valor del atributo `id` se utiliza como nombre de los objetos del complemento Authentic Browser cuando se utilizan en scripts. Por ejemplo, `objPlugIn.SchemaLoadObject.URL` es una llamada al objeto que carga el archivo de esquema. Para más información consulte el apartado [El elemento SCRIPT](#).

style

Este es el atributo HTML `style` normal y corriente y sirve para especificar las dimensiones de la ventana de la vista Authentic en el explorador del cliente.

codebase

El atributo `codebase` aporta la ubicación del archivo CAB. Recuerde que hay un archivo CAB para el complemento de 32 bits (`AuthenticBrowserEdition.CAB`) y otro para el complemento de 64 bits(`AuthenticBrowserEdition_x64.CAB`).

El valor de la extensión opcional `#version` aporta el número de versión del componente que está disponible en el servidor. Si el cliente tiene una versión anterior y en el atributo `codebase` se indica una versión más reciente, se instalará la versión más reciente del servidor. Si no se especifica la extensión `#version`, no se llevará a cabo ninguna actualización hasta que el componente se elimine a mano del cliente. El número de versión actual del componente se enumera junto con las propiedades del archivo DLL del archivo CAB del componente (haga clic con el botón secundario en el archivo y seleccione **Propiedades** en el menú contextual).

classid

Los `classid` para los archivos CAB de 32 y 64 bits son idénticos y son los que aparecen en la tabla del apartado [Versiones de la edición Authentic Browser](#).

A partir de la versión 5.0 del complemento, el valor de `classid` para las versiones Unicode es distinto. Por tanto, si actualiza el archivo CAB Unicode del servidor cuya versión sea anterior a la versión 5.0, asegúrese de cambiar los valores de `classid` en los archivos HTML. También debe tener en cuenta que si un archivo CAB nuevo del servidor tiene el mismo CLSID que un archivo CAB instalado previamente en el cliente, el archivo CAB nuevo no reemplazará automáticamente el antiguo en el cliente. Deberá quitar a mano el archivo CAB instalado antes de descargar el archivo CAB nuevo. Los valores CLSID difieren dependiendo del idioma del complemento Authentic Browser.

Además, al seleccionar la versión que se debe descargar al cliente, debe tener en cuenta que algunos sistemas operativos antiguos no pueden instalar correctamente las versiones Unicode.

A continuación aparecen los valores CLSID de las versiones actuales:

EN Trusted Unicode:

`clsid:B4628728-E3F0-44a2-BEC8-F838555AE780`

EN Untrusted Unicode:

`clsid:A5985EA9-3332-4ddf-AD7F-F6E98BFEAF94`

DE Trusted Unicode:

`cl si d: 91DDF44A- DFD1- 4F47- 8EE3- 4CBE874584F7`

DE Untrusted Unicode:

clsid:28A640E8- EAEE- 4B5D- BEBE- BFA956081E66

Parámetros

El elemento OBJECT puede usar estos parámetros tantas veces como se quiera:

LicServer

El nombre del servidor para el que se registró la licencia de Authentic Browser Enterprise Edition. (Para Authentic Browser Community Edition no hace falta licencia)

LicKey

La licencia para validar el uso de Authentic Browser Enterprise Edition. (Para Authentic Browser Community Edition no hace falta licencia)

LicCompany

El nombre de la compañía con la que se registró la licencia de Authentic Browser Enterprise Edition. (Para Authentic Browser Community Edition no hace falta licencia)

XMLDataURL

Una URL absoluta que aporta la ubicación del archivo XML que se debe editar. En la versión Untrusted también puede usar una ruta de acceso local completa.

XMLDataSaveURL

Una URL absoluta que aporta la ubicación donde se debe guardar el archivo XML. En la versión Untrusted también puede usar una ruta de acceso local completa.

SPSDataURL

Una URL absoluta que aporta la ubicación del archivo SPS. En la versión Untrusted también puede usar una ruta de acceso local completa.

SchemaDataURL

Una URL absoluta que aporta la ubicación del archivo de esquema asociado. En la versión Untrusted también puede usar una ruta de acceso local completa.

TextStateBmpURL

La carpeta donde se deben almacenar las imágenes para los iconos de estado del texto.

TextStateToolbarLine

La línea de la barra de herramientas donde se deben colocar los iconos de estado del texto. Valor predeterminado: 1.

AutoHideUnusedCommandGroups

Indica si se deben ocultar o no los grupos de comandos no utilizados de las barras de herramientas. Valor predeterminado: true.

ToolbarsEnabled

Indica si se habilitan o no las barras de herramientas. Valor predeterminado: true.

ToolbarTooltipsEnabled

Indica si se habilita o no la función de información rápida.

HideSaveButton

Si su valor es True, quita el botón **Guardar** de la barra de herramientas Authentic, que aparece por defecto.

BaseUrl

Aporta la URL base que se debe usar con rutas de acceso relativas.

SaveButtonUsePOST

Si su valor es True, el comando POST HTTP se usa en lugar de PUT a la hora de guardar documentos.

EntryHelpersEnabled

Si su valor es True, se muestran los ayudantes de entrada de Authentic.

EntryHelperSize

El ancho de la ventana del ayudante de entrada medido en píxeles.

EntryHelperAlignment

Especifica la ubicación de los ayudantes de entrada con respecto a la ventana de documento.

- 0 = Alinear la barra de herramientas encima del documento.
- 1 = Alinear la barra de herramientas a la izquierda del documento.
- 2 = Alinear la barra de herramientas debajo del documento.
- 3 = Alinear la barra de herramientas a la derecha del documento.

EntryHelperWindows

Selecciona qué ventanas secundarias del ayudante de entrada se pueden ver.

- 1 = Elementos
- 2 = Atributos
- 4 = Entidades

Puede usar cualquiera de estos valores y en cualquier orden.

SaveButtonAutoEnable

Consulte [Authentic.SaveButtonAutoEnable](#)

LoaderSettingsFileURL

Aporta la URL de LoaderSettingsFile para la gestión de paquetes.

3.2.2 El elemento SCRIPT

Los elementos SCRIPT definen los controladores de eventos y las subrutinas a los que se puede llamar desde el archivo HTML.

Aquí puede ver un script para controlar un evento:

```
<SCRIPT LANGUAGE="javascript" FOR=objPlugIn EVENT="ControlInitialized">
    objPlugIn.SchemaLoadobject.URL = "http://yourserver/OrgChart.xsd"
    objPlugIn.XMLDataLoadobject.URL = "http://yourserver/OrgChart.xml"
    objPlugIn.DesignDataLoadobject.URL = "http://yourserver/OrgChart.sps"
    objPlugIn.StartEditing
</SCRIPT>
```

Y aquí puede ver otro script que contiene subrutinas:

```
<SCRIPT ID=clientEventHandlers LANGUAGE=vbscript>
    Sub BtnOnClick
        objPlugIn.SchemaLoadObject.URL = "http://yourserver/orgChart.xsd"
        objPlugIn.XMLDataLoadObject.URL = "http://yourserver/orgChart.xml"
        objPlugIn.DesignDataLoadObject.URL = "http://yourserver/orgChart.sps"
        objPlugIn.StartEditing
    End Sub
    Sub OnClickFind
        objPlugIn.FindDialog
    End Sub
    Sub BtnOnTestProp
        If objPlugIn.IsRowInsertEnabled Then
            msgbox "true"
        Else
            msgbox "false"
        End If
    End Sub
</SCRIPT>
```

Lenguaje

El complemento Authentic Browser se ha probado con JavaScript y VBScript.

Controlar eventos

El valor del atributo ID del elemento OBJECT del cuerpo HTML se especifica como valor del atributo FOR. Los objetos del complemento Authentic Browser a los que se llama deben tener un nombre que es este valor.

Para ver una lista con todos los eventos consulte el apartado [Eventos de la Referencia](#).

Subrutinas

Puede crear subrutinas para cualquier evento que quiera definir en el archivo HTML. El nombre del objeto de Authentic Browser debe ser el mismo que el valor del atributo ID del elemento OBJECT del cuerpo HTML. En el script anterior, por ejemplo, el prefijo es objPlugIn, que debe ser el valor del atributo ID del elemento OBJECT.

Los métodos, propiedades y subobjetos disponibles en el complemento Authentic Browser se describen en la *Referencia* de este manual.

3.2.3 Ejemplo nº1: ejemplo básico

El código HTML que aparece más abajo genera una página con estas características:

- Instala la versión Trusted Unicode de Authentic Browser en el cliente (si no está instalado todavía).
- La página contiene una ventana de 600 x 500 píxeles en la que se carga Authentic Browser.
- Debajo de la ventana de Authentic Browser hay una fila de cuatro botones.
- Se carga la vista Authentic del archivo `OrgChart.xml`.
- Los botones **Buscar** y **Reemplazar** abren los cuadros de diálogo "Buscar" y "Reemplazar" respectivamente.
- El botón **Guardar** guarda los cambios en un archivo llamado `SaveFile.xml` ubicado en el directorio raíz del servidor.
- El botón **Probar propiedad** prueba una propiedad simple.

Cuando se abre esta página HTML en el cliente, el usuario puede empezar a editar el archivo XML `OrgChart.xml` y guardar el archivo editado como `SaveFile.xml`.

Si quiere, puede usar esta sencilla página HTML para probar si Authentic Browser funciona correctamente. Antes debe asegurarse de utilizar las URL correctas para ubicar el archivo CAB, los archivos XSD, XML y SPS y cualquier otro recurso en el servidor. Y recuerde que algunos servidores distinguen entre mayúsculas y minúsculas, así que si algún archivo no se encuentra, revise el uso de mayúsculas y minúsculas en los nombres de archivo y en los comandos del código. Este ejemplo se puede modificar o ampliar para desarrollar soluciones más complejas con ayuda de Authentic Browser. También puede consultar el apartado [El elemento OBJECT](#) para obtener más información.

```
<html>
<head>
<meta http-equiv="Content-Type" content="text/html; charset=windows-1252">
<title>Página XMLSpyDocEditPlugIn básica</title>

<!-- Script para controlar el evento ControlInitialized --&gt;
&lt;SCRIPT LANGUAGE="javascript" FOR="objPlugIn" EVENT="ControlInitialized"&gt;
    objPlugIn.SchemaLoadObject.URL = "http://suservidor/orgChart.xsd"
    objPlugIn.XMLDataLoadObject.URL = "http://suservidor/orgChart.xml"
    objPlugIn.DesignDataLoadObject.URL = "http://suservidor/orgChart.sps"
    objPlugIn.StartEditing()
&lt;/SCRIPT&gt;

<!-- Script con subrutas --&gt;
&lt;SCRIPT ID=clientEventHandlers LANGUAGE=vbscript&gt;
    Sub OnClickFind
        objPlugIn.FindDialog
    End Sub

    Sub OnClickReplace
        objPlugIn.ReplaceDialog
    End Sub

    Sub BtnOnSave
        objPlugIn.XMLDataSaveUrl = "http://suservidor/SaveFile.xml"
    End Sub
&lt;/SCRIPT&gt;</pre>
```

```
        objPlugIn.Save
    End Sub

    Sub BtnOnTestProp
        If objPlugIn.IsRowInsertEnabled Then
            msgbox "true"
        Else
            msgbox "false"
        End If
    End Sub
</SCRIPT>
</head>

<body>
    <!-- El elemento Object tiene un atributo id con el valor que se debe usar
-->
    <!-- como nombre de los objetos de Authentic Browser -->
    <!-- Classid selecciona la versión Trusted Unicode -->
    <OBJECT id="objPlugIn"
        <!-- CodeBase selecciona el archivo CAB de 32 bits
        (AuthenticBrowserEdition.CAB) -->
        <!-- o el archivo CAB de 64 bits (AuthenticBrowserEdition_x64.CAB) -->
        CodeBase="http://suservidor/AuthenticBrowserEdition.CAB#Version=12,3,0,0"
        <!-- El atributo Classid para el archivo CAB de 32 bits y para el de 64 bits
        es el mismo -->
        Classid="clsid:23B503E7-269B-45CE-BAB2-22AA97BED8E2" width="600"
    height="500">
        </OBJECT>
        <p>
            <input type="button" value="Find" name="B4" onclick="onClickFind()">
            <input type="button" value="Replace" name="B5" onclick="onClickReplace()">
            <input type="button" value="Save" name="B6" onclick="BtnOnSave()">
            <input type="button" value="Test property" name="B7"
                onclick="BtnOnTestProp">
        </p>
    </body>
</html>
```

3.2.4 Ejemplo nº2: ordenar una tabla

A continuación aparece un ejemplo de página HTML que tiene un JavaScript incrustado y que requiere tener instalado Authentic Browser (archivo CAB) en el equipo. Recuerde que algunos servidores distinguen entre el uso de mayúsculas y minúsculas, así que si algún archivo no se encuentra, revise el uso de mayúsculas y minúsculas en los nombres de archivo y en los comandos del código.

El código del ejemplo:

- Muestra cómo acceder al complemento. Modifique el código para remitir al archivo CAB y al identificador de clase (CLSID) de su versión del complemento (Trusted o Untrusted).
- Muestra cómo cargar un archivo en el complemento. Modifique el código para remitir a su documento de muestra.
- Implementa botones para el posicionamiento del cursor.
- Implementa comandos más complejos como comandos para ordenar tablas.
- Muestra cómo usar el evento SelectionChanged.

Consulte también el apartado [El elemento OBJECT](#) para obtener más información.

```
<html>
  <head>
    <meta http-equiv="Content-Type" content="text/html; charset=windows-1252">
    <title>Página de prueba para Authentic Browser</title>

    <SCRIPT LANGUAGE="javascript" For="objPlugIn" EVENT="ControlInitialized">
      var strSampleRoot = "http://miRaíz/miRuta/miNombreDocumento";
      objPlugIn.SchemaLoadObject.URL = strSampleRoot + ".xsd";
      objPlugIn.XMLDataLoadObject.URL = strSampleRoot + ".xml";
      objPlugIn.DesignDataLoadObject.URL = strSampleRoot + ".sps";
      objPlugIn.StartEditing();
    </SCRIPT>

    <SCRIPT ID="clientEventHandlers" LANGUAGE="javascript">
      var objCurrentRange = Null;

      Function BtnDocumentBegin()
      { objPlugIn.Authenticview.DocumentBegin.Select(); }
      Function BtnDocumentEnd()
      { objPlugIn.Authenticview.DocumentEnd.Select(); }
      Function BtnWholeDocument()
      { objPlugIn.Authenticview.WholeDocument.Select(); }
      Function BtnSelectNextWord()
      { objPlugIn.Authenticview.Selection.SelectNext(1).Select(); }
      Function BtnSortDepartmentOnClick()
      {
        var objCursor = Null;
        var objTableStart = Null;
        var objBubble = Null;
        var strField1 = "";
        var strField1 = "";
        var nColIndex = 0;
        var nRowIndex = 0;
      }
    </SCRIPT>
```

```

objCursor = objPlugIn.AuthenticView.Selection;
If (objCursor.IsInDynamicTable())
{
    // calcular índice de columnas actual
    nColIndex = 0;
    While (True)
    {
        try { objCursor.GotoPrevious(11); }
        catch (err) { break; }
        nColIndex++;
    }

    // Ir al principio de la tabla
    objTableStart = objCursor.ExpandTo(9).CollapsToBegin().Clone();

    // Recuento de filas
    nRows = 1;
    While (True)
    {
        try { objTablestart.GotoNext(10); }
        catch (err) { break; }
        nRows++;
    }

    // Ordenación de burbuja
    For (var i = 0; i < nRows - 1; i++) {
        for(var j = 0; j < nRows-i-1; j++) {
            objBubble = objCursor.ExpandTo(9).CollapsToBegin().Clone();
            // seleccionar columna correcta en la fila de tabla nºX

            objBubble.GotoNext(6).Goto(10,j,2).Goto(11,nColIndex,2).ExpandTo(6);
            strField1 = objBubble.Text;
            strField2 =
            objBubble.GotoNext(10).Goto(11,nColIndex,2).ExpandTo(6).Text;
            if(strField1 > strField2) {
                if(!objBubble.MoveRowUp()) {
                    alert('No está permitido mover la fila de la
tabla.');
                    return;
                }
            }
        }
    }
}

</SCRIPT>
</head>

<body>
<Object id="objPlugIn"
<!-- CodeBase selecciona el archivo CAB de 32 bits
(AuthenticBrowserEdition.CAB) -->
<!-- o el archivo CAB de 64 bits (AuthenticBrowserEdition_x64.CAB) -->
codeBase="http://miArchivoCAB/
AuthenticBrowserEdition.CAB#Version=12,3,0,0">

```

```
<!-- El atributo classid para el archivo CAB de 32 bits y para el de 64  
bits es el mismo -->  
    classid="clsid:23B503E7-269B-45CE-BAB2-22AA97BED8E2"  
    width="100%"  
    height="80%"  
    VIEWASTEXT>  
    <PARAM NAME="EntryHelpersEnabled" VALUE="TRUE">  
    <PARAM NAME="SaveButtonAutoEnable" VALUE="TRUE">  
  </Object>  
  <TABLE>  
    <TR>  
      <TD><Input Type="button" value="Ir al principio" id="B1"  
        onclick="BtnDocumentBegin()"></TD>  
      <TD><Input Type="button" value="Ir al final" name="B2"  
        onclick="BtnDocumentEnd()"></TD>  
      <TD><Input Type="button" value="Documento entero" name="B3"  
        onclick="BtnwholeDocument()"></TD>  
      <TD><Input Type="button" value="Seleccionar siguiente palabra"  
        name="B4" onclick="BtnSelectNextWord()"></TD>  
    </TR>  
    <TR>  
      <TD><Input Type="button" value="Ordenar tabla en función de esta  
        columna" id="B6" onclick="BtnsSortDepartmentOnClick()"></TD>  
    </TR>  
  </TABLE>  
  <TABLE id=SelTable border=1>  
    <TR><TD id=SelTable_FirstTextPosition></TD><TD  
      id=SelTable_LastTextPosition></TD></TR>  
    <TR><TD id=SelTable_FirstXMLData></TD><TD  
      id=SelTable_FirstXMLDataOffset></TD></TR>  
    <TR><TD id=SelTable_LastXMLData></TD><TD  
      id=SelTable_LastXMLDataOffset></TD></TR>  
    <TR><TD id=SelTable_Text></TD></TR>  
  </TABLE>  
</body>  
  
<SCRIPT LANGUAGE=javascript For=objPlugIn EVENT=selectionchanged>  
  var CurrentSelection = Null;  
  CurrentSelection = objPlugIn.Authenticview.Selection;  
  SelTable_FirstTextPosition.innerHTML = CurrentSelection.FirstTextPosition;  
  SelTable_LastTextPosition.innerHTML = CurrentSelection.LastTextPosition;  
  SelTable_FirstXMLData.innerHTML =  
  CurrentSelection.FirstXMLData.Parent.Name;  
  SelTable_FirstXMLDataOffset.innerHTML =  
  CurrentSelection.FirstXMLDataOffset;  
  SelTable_LastXMLData.innerHTML = CurrentSelection.LastXMLData.Parent.Name;  
  SelTable_LastXMLDataOffset.innerHTML = CurrentSelection.LastXMLDataOffset;  
</SCRIPT>  
</html>
```

3.3 Firefox

Si la página HTML para el complemento de Authentic se abre en Firefox, debe incluir estos elementos:

- Un elemento HTML [EMBED](#) que (i) provoca la descarga del DLL para el complemento de Authentic desde el servidor al cliente y (ii) especifica las dimensiones de la ventana de la vista Authentic en el explorador del cliente. El elemento [EMBED](#) contiene la ubicación del complemento Authentic Browser.
- Uno o varios [agentes de escucha de eventos](#) para definir las subrutinas y el control de eventos. Los agentes de escucha de eventos se construyen dentro de elementos SCRIPT.

Este apartado tiene varios subapartados:

- [El elemento EMBED](#): describe cómo se utiliza el elemento HTML [EMBED](#) en una página HTML para el complemento de Authentic.
- [Añadir agentes de escucha de eventos](#): describe cómo se utilizan los [agentes de escucha de eventos](#) en una página HTML para el complemento de Authentic.
- Ejemplos de páginas HTML completas: [Firefox. Ejemplo nº1: ejemplo básico](#) y [Firefox. Ejemplo nº2: ordenar una tabla](#).

Si necesita información sobre algún objeto en particular consulte el apartado [Objetos](#) de la [Referencia](#).

Notas:

- El complemento Authentic Browser es compatible con [Firefox](#).
- Compruebe que el tipo MIME para archivos XPI y CRX se [añadió en el servicio de explorador](#) del servidor en la lista de tipos MIME para el sitio que usará.

3.3.1 El elemento EMBED

El elemento `EMBED` cumple estas funciones:

- Pone un nombre al complemento Authentic Browser (a través del atributo `id`).
- Selecciona qué [versión del complemento Authentic Browser](#) se debe utilizar (a través del atributo `type`).
- Especifica un archivo XPI o CRX (que es el DLL del complemento) a través del atributo `PluginsPage`. El valor del atributo `PluginsPage` es una ruta de acceso que encuentra el archivo XPI o CRX.
- Especifica la dimensión de la ventana de la vista Authentic en el explorador del cliente (a través de los atributos `height` y `width`).

Más abajo puede ver un ejemplo de elemento HTML `EMBED`. Este elemento selecciona la versión Trusted Unicode (con el valor dado en el atributo `type`) y establece unas dimensiones de 100% y 60% para la ventana de la vista Authentic en el explorador del cliente.

`id`

El valor del atributo `id` se utiliza como nombre de los objetos del complemento Authentic Browser cuando se utilizan en scripts. Por ejemplo, `objPlugIn.SchemaLoadObject.URL` es una llamada al objeto que carga el archivo de esquema.

`type`

Este valor es el [tipo MIME](#) de la [versión correspondiente del complemento Authentic Browser](#). En el ejemplo que aparece más abajo el valor `application/x-authentic-scriptable-plugin` identifica la versión Trusted. Para ver la lista de versiones y sus tipos MIME consulte el apartado [Versiones de la edición Authentic Browser](#).

`width, height`

Estos atributos especifican las dimensiones de la ventana de la vista Authentic que se debe crear en la ventana del explorador.

`PluginsPage`

El valor de este atributo especifica la ubicación del archivo XPI (Firefox) del complemento Authentic Browser en el servidor. Compruebe que usa la ruta de acceso correcta en la URL que busca el archivo XPI. Algunos servidores distinguen entre mayúsculas y minúsculas, así que si algún archivo no se encuentra, revise el uso de mayúsculas y minúsculas en los nombres de archivo y en las rutas de acceso.

`LicServer`

El nombre del servidor para el que se registró la licencia de Authentic Browser Enterprise Edition. (Para Authentic Browser Community Edition no hace falta licencia)

`LicKey`

La licencia para validar el uso de Authentic Browser Enterprise Edition. (Para Authentic Browser Community Edition no hace falta licencia)

`LicCompany`

El nombre de la compañía con la que se registró la licencia de Authentic Browser Enterprise Edition. (Para Authentic Browser Community Edition no hace falta licencia)

Parámetros

Los parámetros (*ver lista más abajo*) se pueden usar de varias maneras:

- Dando el nombre del parámetro y su valor como par atributo/valor del elemento EMBED. Por ejemplo, el parámetro ToolbarsEnabled puede definirse con el valor true añadiendo el par atributo/valor ToolbarsEnabled="true" en el elemento EMBED.
- Puede especificar elemento PARAM como secundario del elemento OBJECT:

```
<object id="objPlugIn"
        type="application/x-authentic-scriptable-plugin"
        width="100%"
        height="60%">
    <param name="ToolbarsEnabled"
          value="true"/>
</object>
```

Sin embargo, hay un inconveniente si los parámetros se especifican de esta forma, es decir, como secundario del elemento OBJECT. Como Firefox no acepta el atributo PLUGINSPAGE del elemento OBJECT, no tendrá ninguna referencia al archivo XPI de Authentic Browser del servidor y no podrá iniciar la instalación del complemento Authentic Browser en el cliente. Por tanto, solo debería especificar parámetros de esta manera si el complemento ya está instalado.

XMLDataURL

Una URL absoluta que aporta la ubicación del archivo XML que se debe editar. En la versión Untrusted también puede usar una ruta de acceso local completa.

XMLDataSaveURL

Una URL absoluta que aporta la ubicación donde se debe guardar el archivo XML. En la versión Untrusted también puede usar una ruta de acceso local completa.

SPSDataURL

Una URL absoluta que aporta la ubicación del archivo SPS. En la versión Untrusted también puede usar una ruta de acceso local completa.

SchemaDataURL

Una URL absoluta que aporta la ubicación del archivo de esquema asociado. En la versión Untrusted también puede usar una ruta de acceso local completa.

TextStateBmpURL

La carpeta donde se deben almacenar las imágenes para los iconos de estado del texto.

TextStateToolbarLine

La línea de la barra de herramientas donde se deben colocar los iconos de estado del texto. Valor predeterminado: 1.

AutoHideUnusedCommandGroups

Indica si se deben ocultar o no los grupos de comandos no utilizados de las barras de herramientas. Valor predeterminado: true.

ToolbarsEnabled

Indica si se habilitan o no las barras de herramientas. Valor predeterminado: true.

ToolbarToolipsEnabled

Indica si se habilita o no la función de información rápida.

HideSaveButton

Si su valor es True, quita el botón **Guardar** de la barra de herramientas Authentic, que aparece por defecto.

BaseUrl

Aporta la URL base que se debe usar con rutas de acceso relativas.

SaveButtonUsePOST

Si su valor es True, el comando POST HTTP se usa en lugar de PUT a la hora de guardar documentos.

EntryHelpersEnabled

Si su valor es True, se muestran los ayudantes de entrada de Authentic.

EntryHelperSize

El ancho de la ventana del ayudante de entrada medido en pixels.

EntryHelperAlignment

Especifica la ubicación de los ayudantes de entrada con respecto a la ventana de documento.

0 = Alinear la barra de herramientas encima del documento.

1 = Alinear la barra de herramientas a la izquierda del documento.

2 = Alinear la barra de herramientas debajo del documento.

3 = Alinear la barra de herramientas a la derecha del documento.

EntryHelperWindows

Selecciona qué ventanas secundarias del ayudante de entrada se pueden ver.

1 = Elementos

2 = Atributos

4 = Entidades

Puede usar cualquiera de estos valores y en cualquier orden.

SaveButtonAutoEnable

Consulte [Authentic.SaveButtonAutoEnable](#)

LoaderSettingsFileURL

Aporta la URL de LoaderSettingsFile para la gestión de paquetes.

3.3.2 Añadir agentes de escucha de eventos

El elemento `SCRIPT` que aparece a continuación define un agente de escucha de eventos y lo registra con el objeto del complemento. La función del agente de escucha se invocará cada vez que el evento se desencadene dentro del complemento.

```
<SCRIPT LANGUAGE="javascript">
var selCount = 0;
function onSelectionChanged()
{
    selCount = selCount + 1;
    selectionCounter.value = "SelectionCount = " + selCount;
}
var objPlugIn = document.getElementById('objPlugIn');
objPlugIn.addEventListener("selectionchanged", onSelectionChanged, false)
</SCRIPT>
```

Para ver una lista de eventos consulte el apartado [Eventos de la Referencia](#).

Lenguaje

El complemento Authentic Browser se ha probado con JavaScript y VBScript.

Agentes de escucha de eventos

Para más información consulte la [recomendación correspondiente del W3C](#).

Modelo de objetos de Authentic Browser

Los métodos, las propiedades y los subobjetos disponibles en el complemento Authentic Browser se describen en la sección [Referencia](#).

3.3.3 Ejemplo nº1: ejemplo básico

El código HTML que aparece más abajo genera una página con estas características:

- Instala la versión Trusted de Authentic Browser para Firefox en el cliente (si no está instalado todavía).
- La ventana del complemento dentro de la página tiene un ancho que es 100% de la ventana del explorador y 60% de su alto.
- Debajo de la ventana de Authentic Browser hay una fila de cuatro botones.
- Se carga la vista Authentic del archivo OrgChart.xml.
- Los botones **Buscar** y **Reemplazar** abren los cuadros de diálogo "Buscar" y "Reemplazar" respectivamente.
- El botón **Guardar** guarda los cambios en un archivo llamado SaveFile.xml ubicado en el directorio raíz del servidor.
- El botón **Probar propiedad** prueba una propiedad simple.

Cuando se abre esta página HTML en el cliente, el usuario puede empezar a editar el archivo XML orgChart.xml y guardar el archivo editado como SaveFile.xml.

Si quiere, puede usar esta sencilla página HTML para probar si Authentic Browser funciona correctamente. Antes debe asegurarse de utilizar la dirección IP y la ruta de acceso correctas de los archivos correspondientes en las URL que buscan el archivo XPI, los archivos XSD, XML y SPS y cualquier otro recurso en el servidor. Y recuerde que algunos servidores distinguen entre mayúsculas y minúsculas, así que si algún archivo no se encuentra, revise el uso de mayúsculas y minúsculas en los nombres de archivo y en los comandos del código. Este ejemplo se puede modificar o ampliar para desarrollar soluciones más complejas con ayuda de Authentic Browser.

```
<html>
<head>
    <meta http-equiv="Content-Type" content="text/html; charset=windows-1252">
        <title>Página básica para Authentic Browser</title>
</head>
<!-- to disable the fast-back cache in Firefox, define an unload handler -->
<BODY id="bodyId" onunload="Unload()">
    <!-- Embed element has id with value that must be used -->
    <!-- as name of Authentic Browser Plug-in objects -->
    <!-- type selects the Trusted Unicode version -->
    <embed
        id="objPlugIn"
        type="application/x-authentic-scriptable-plugin"
        width="100%"
        height="60%"
        PLUGINSPAGE="http://your-server-including-path/
AuthenticFirefoxPlugin_trusted.xpi"
        LicServer="DevAuthBrowTest"
        LicCompany="Altova"
        LicKey="xxxxxxxx" />
    <!-- Script with subroutines -->
    <SCRIPT LANGUAGE="javascript">
        var objPlugIn = document.getElementById('objPlugIn');
        function onclickFind()
        {

```

```

        objPlugIn.FindDialog();
    }

    function onClickReplace()
    {
        objPlugIn.ReplaceDialog();
    }

    function BtnOnSave()
    {
        objPlugIn.XMLDataSaveUrl = "http://su-servidor/Authentic/
SaveFile.xml"
        objPlugIn.Save()
    }

    function BtnOnTestProp()
    {
        alert ( objPlugIn.IsRowInsertEnabled );
    }
    function Unload()
    {
    }
    function InitAuthenticPluginPage( )
    {
        var serverstr=su-servidor '/';
        var basedir='Authentic/';
        objPlugIn.SchemaLoadObject.URL = 'http://' + serverstr + basedir +
'OrgChart.xsd';
        objPlugIn.XMLDataLoadObject.URL = 'http://' + serverstr + basedir +
'OrgChart.xml' ;
        objPlugIn.DesignDataLoadobject.URL = 'http://' + serverstr +
basedir + 'OrgChart.sps';
        objPlugIn.StartEditing();
    }
    // event subscription if running on Firefox
    objPlugIn.addEventListener("ControlInitialized", InitAuthenticPluginPage,
false);
</SCRIPT>
<p>
<input type="button" value="Find" name="B4" onclick="onClickFind()">
<input type="button" value="Replace" name="B5" onclick="onClickReplace()">
<input type="button" value="Save" name="B6" onclick="BtnOnSave()">
<input type="button" value="Test property" name="B7"
onclick="BtnOnTestProp()">
</p>
</body>
</html>

```

Nota: el script anterior contiene la información sobre la licencia necesaria para activar Authentic Browser Enterprise Edition. Si faltan los tres parámetros LicServer, LicCompany y LicKey, entonces solamente se habilitarán las funciones de la edición Community Edition.

3.3.4 Ejemplo nº2: ordenar una tabla

A continuación aparece un ejemplo de página HTML que tiene un JavaScript incrustado y que requiere tener instalado Authentic Browser (archivo XPI) en el equipo. Recuerde que algunos servidores distinguen entre el uso de mayúsculas y minúsculas, así que si algún archivo no se encuentra, revise el uso de mayúsculas y minúsculas en los nombres de archivo y en los comandos del código.

El código del ejemplo:

- Muestra cómo acceder al complemento. Modifique el código para remitir al archivo XPI y al identificador de clase (tipo MIME) de su versión del complemento (Trusted o Untrusted).
- Muestra cómo cargar un archivo en el complemento. Modifique el código para remitir a su documento de muestra.
- Implementa botones para el posicionamiento del cursor.
- Implementa comandos más complejos como comandos para ordenar tablas.
- Muestra cómo usar el evento SelectionChanged.

```
<html>
<head>
<meta http-equiv="Content-Type" content="text/html; charset=windows-1252">
<title>Página de prueba para Authentic Browser</title>

</head>
<!-- to disable the fast-back cache in Firefox, define an unload handler -->
<BODY id="bodyId" onunload="unload()">
<embed
    id="objPlugIn"
    type="application/x-authentic-scriptable-plugin"
    width="100%"
    height="60%"
    PLUGINSPAGE="http://tu-servidor-incluida-la-ruta/
AuthenticFirefoxPlugin_trusted.xpi"
    EntryHelpersEnabled="TRUE"
    SaveButtonAutoEnable="TRUE" >
</embed>
<TABLE>
<SCRIPT LANGUAGE="javascript">
var objCurrentRange = null;
var objPlugIn = document.getElementById('objPlugIn');

function BtnDocumentBegin() { objPlugIn.AuthenticView.DocumentBegin.Select(); }
function BtnDocumentEnd() { objPlugIn.AuthenticView.DocumentEnd.Select(); }
function BtnwholeDocument() { objPlugIn.AuthenticView.wholeDocument.Select(); }
function BtnSelectNextword()
{ objPlugIn.AuthenticView.Selection.SelectNext(1).Select(); }

function BtnSortDepartmentonClick()
{
    var objCursor = null;
    var objTableStart = null;
    var objBubble = null;
    var strField1 = "";
    var strField1 = "";
    var nColIndex = 0;
```

```

var nRows = 0;

objCursor = objPlugIn.AuthenticView.Selection;
if (objCursor.IsInDynamicTable())
{
    // calcular índice de columnas actual
    nColIndex = 0;
    bContinue = true;
    while ( bContinue )
    {
        try { objcursor.GotoPrevious(11); }
        catch (err) { bContinue = false; nColIndex--; }
        nColIndex++;
    }

    // Ir al principio de la tabla
    objTableStart = objCursor.ExpandTo(9).CollapsToBegin().Clone();

    // recuento de filas
    nRows = 1;
    bContinue = true;
    while ( bContinue )
    {
        try { objTableStart.GotoNext(10); }
        catch (err) { bContinue = false; }
        nRows++;
    }

    // ordenación de burbuja
    for ( i = 0; i < nRows - 1; i++ ) {
        for( j = 0; j < nRows-i-1; j++) {
            objBubble =
objCursor.ExpandTo(9).CollapsToBegin().Clone();
                                // Seleccionar columna correcta en la fila de tabla
nºX

            objBubble.GotoNext(6).Goto(10,j,2).Goto(11,nColIndex,2).ExpandTo(6);
            strField1 = objBubble.Text;
            try
            {
                strField2 =
objBubble.GotoNext(10).Goto(11,nColIndex,2).ExpandTo(6).Text;
            }
            catch ( err ) { continue; };
            if(strField1 > strField2) {
                if(!objBubble.MoveRowUp()) {
                    alert(No está permitido mover la
fila.');
                    return;
                }
            }
        }
    }
}

function InitAuthenticPluginPage( )

```

```
{  
    var serverstr=su-servidor/';  
    var basedir='Authentic/';  
    objPlugIn.SchemaLoadObject.URL = 'http://' + serverstr + basedir +  
    'OrgChart.xsd';  
    objPlugIn.XMLDataLoadObject.URL = 'http://' + serverstr + basedir +  
    'OrgChart.xml';  
    objPlugIn.DesignDataLoadObject.URL = 'http://' + serverstr + basedir +  
    'OrgChart.sps';  
    objPlugIn.StartEditing();  
}  
  
function Unload()  
{  
}  
  
function OnSelectionChanged()  
{  
    var CurrentSelection = null;  
    CurrentSelection = objPlugIn.AuthenticView.Selection;  
    SelTable_FirstTextPosition.innerHTML =  
    CurrentSelection.FirstTextPosition;  
    SelTable_LastTextPosition.innerHTML = CurrentSelection.LastTextPosition;  
    SelTable_FirstXMLData.innerHTML =  
    CurrentSelection.FirstXMLData.Parent.Name;  
    SelTable_FirstXMLDataOffset.innerHTML =  
    CurrentSelection.FirstXMLDataOffset;  
    SelTable_LastXMLData.innerHTML =  
    CurrentSelection.LastXMLData.Parent.Name;  
    SelTable_LastXMLDataOffset.innerHTML =  
    CurrentSelection.LastXMLDataOffset;  
}  
  
objPlugIn.addEventListener("selectionchanged", OnSelectionChanged, false)  
// suscripción a evento si se ejecuta en Firefox  
objPlugIn.addEventListener("ControlInitialized", InitAuthenticPluginPage,  
false);  
</SCRIPT>  
<TR><TD><input type="button" value="Ir al principio" id="B1"  
onclick="BtnDocumentBegin()"/></TD>  
<TD><input type="button" value="Ir al final" name="B2"  
onclick="BtnDocumentEnd()"/></TD>  
<TD><input type="button" value="Documento entero" name="B3"  
onclick="BtnwholeDocument()"/></TD>  
<TD><input type="button" value="Seleccionar palabra siguiente" name="B4"  
onclick="BtnSelectNextword()"/></TD>  
</TR>  
<TR>  
<TD><input type="button" value="Ordenar tabla en función de esta columna"  
id="B6" onclick="BtnSortDepartmentonClick()"/></TD>  
</TR>  
</TABLE>  
<TABLE id=SelTable border=1>  
<TR><TD id=SelTable_FirstTextPosition></TD><TD id=SelTable_LastTextPosition></TD></TR>  
<TR><TD id=SelTable_FirstXMLData></TD><TD id=SelTable_FirstXMLDataOffset></TD></TR>  
<TR><TD id=SelTable_LastXMLData></TD><TD id=SelTable_LastXMLDataOffset></TD></TR>  
<TR><TD id=SelTable_Text></TD></TR>  
</TABLE>  
</body>  
</html>
```

3.4 Para cualquier explorador

En algunos casos no se conoce qué explorador (IE o Firefox) se usará en el cliente. En casos así se pueden almacenar versiones de [Authentic Browser para IE y para Firefox](#) en el servidor (es decir, tanto el archivo CAB como el archivo XPI). En la página HTML puede insertar un script para determinar qué explorador se utilizó para abrir la página HTML y provocar la carga del complemento [Authentic Browser](#) correspondiente.

Además, si el explorador utilizado en el cliente es IE, debe seleccionarse y descargarse el archivo .CAB correcto (para IE de 32 o 64 bits). Puede usar un script para determinar la versión de IE y seleccionar el archivo .CAB correspondiente.

Este apartado contiene un [ejemplo](#) que determina el tipo de explorador, carga la versión correcta del complemento Authentic Browser y lleva a cabo algunas funciones.

Si necesita información sobre algún objeto en particular consulte el apartado [Objetos](#) de la [Referencia](#).

Notas:

- El complemento Authentic Browser es compatible con **Internet Explorer 5.5 o superior y Mozilla Firefox**.
- Si va a trabajar con Firefox, compruebe que el tipo MIME para archivos XPI se [añadió en el servicio de explorador](#) del servidor en la lista de tipos MIME para el sitio que usará.

3.4.1 Ejemplo para cualquier explorador

El código HTML que aparece más abajo genera una página con estas características:

- Comprueba qué explorador web está instalado en el cliente (IE o Firefox) e instala una versión de Authentic Browser para el tipo de explorador detectado.
- Si el explorador detectado es IE, comprueba si el sistema es de 32 o 64 bits y selecciona el [correspondiente archivo CAB](#).
- La ventana de Authentic Browser dentro de la página tiene un ancho que es 100% de la ventana del explorador y 60% de su alto.
- Debajo de la ventana de Authentic Browser hay una fila de cinco botones.
- El botón **Iniciar edición** carga la vista Authentic del archivo `OrgChart.xml`, que está en el directorio raíz del servidor.
- Los botones **Buscar** y **Reemplazar** abren los cuadros de diálogo "Buscar" y "Reemplazar" respectivamente.
- El botón **Guardar** guarda los cambios en un archivo llamado `SaveFile_OrgChart.xml` ubicado en el directorio raíz del servidor.
- El botón **Probar propiedad** prueba una propiedad simple.

Cuando se abre esta página HTML en el cliente, el usuario puede empezar a editar el archivo XML `OrgChart.xml` y guardar el archivo editado como `SaveFile_OrgChart.xml`.

Si quiere, puede usar esta sencilla página HTML para probar si Authentic Browser funciona correctamente. Antes debe asegurarse de utilizar la dirección IP y la ruta de acceso correctas de los archivos correspondientes en las URL que buscan el archivo XPI, los archivos XSD, XML y SPS y cualquier otro recurso en el servidor. Y recuerde que algunos servidores distinguen entre mayúsculas y minúsculas, así que si algún archivo no se encuentra, revise el uso de mayúsculas y minúsculas en los nombres de archivo y en los comandos del código. Este ejemplo se puede modificar o ampliar para desarrollar soluciones más complejas con ayuda de Authentic Browser.

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN" "http://www.w3.org/TR/html4/loose.dtd">
<html>
  <head>
    <meta http-equiv="Content-Type" content="text/html; charset=UTF-8">
    <title>Página de prueba para Orgchart.sps compatible con cualquier explorador</title>
    <script type="text/javascript">
      <!--
        function BtnOnSave() { objPlugIn.Save(); }

        function InitAuthenticPluginPage( )
        {

          var schema= document.getElementById('xsd');
          var instance=document.getElementById('xml');
          var design=document.getElementById('sps');
          objPlugIn.XMLDataLoadObject.URL =instance.innerHTML;
          objPlugIn.DesignDataLoadObject.URL = design.innerHTML;
          objPlugIn.SchemaLoadObject.URL= schema.innerHTML;
        }
      -->
    </script>
  </head>
  <body>
    <div id="OrgChart"></div>
  </body>
</html>
```

```

        // alert(schema.innerHTML+" "+instance.innerHTML+" "
+design.innerHTML);

        /*
         var serverstr='su-servidor/';
         var basedir='Authentic/';
         objPlugIn.SchemaLoadObject.URL = 'http://' + serverstr + basedir +
'OrgChart.xsd';
         objPlugIn.XMLDataLoadObject.URL = 'http://' + serverstr + basedir +
'OrgChart.xml' ;
         objPlugIn.DesignDataLoadObject.URL = 'http://' + serverstr + basedir +
'OrgChart.sps';
        */

        objPlugIn.StartEditing();

    }

    function Unload()
    {
    }
    //-->
</script>
<style type="text/css">@page { margin-left:0.60in; margin-right:0.60in;
margin-top:0.79in; margin-bottom:0.79in } @media screen { br.altova-page-
break { display: none; } } @media print { br.altova-page-break { page-break-
before: always; } }
</style>
</head>

<body id="bodyId" onunload="Unload()">
<table border="1">
<tbody>
    <tr><th><span>DesignLoadURL</span></th><td id="sps">http://su-
servidor/Authentic/Orgchart.sps</td></tr>
    <tr><th><span>SchemaLoadURL</span></th><td id="xsd">http://su-
servidor/Authentic/Orgchart.xsd</td></tr>
    <tr><th><span>XMLDataLoadURL</span></th><td id="xml">http://su-
servidor/Authentic/Orgchart.xml</td></tr>
    <tr><th><span>XMLDataSaveURL</span></th><td id="xmlsave">http://su-
servidor/Authentic/SaveFile_OrgChart.xml</td></tr>
</tbody>
</table>
<center><h3><span>Authentic Platform Independent Plug-in Enterprise
Edition</span></h3></center>
<span>&ampnbsp</span>
<center>
<script language="JavaScript" type="text/javascript">
// devolver true si la página se carga en Firefox
function isFirefoxOnWindows()
{
    return ((navigator.userAgent.indexOf('Firef') != -1) &&
(navigator.userAgent.indexOf('win') != -1));
}

// devolver true si la página se carga en Internet Explorer

```

```
function isIEOnWindows()
{
    return ((navigator.userAgent.indexOf('MSIE') != -1) &&
(navigator.userAgent.indexOf('Win') != -1))
}

//devolver true si el explorador es de 64 bits
function is64bitBrowser()
{
    return ((navigator.userAgent.indexOf('Win64') != -1)&&
(navigator.userAgent.indexOf('x64') != -1))
}

//devolver Codebase para versión de 32 o 64 bits
function getCodeBase()
{
    if ( is64bitBrowser() ){
        return('CodeBase="http://su-servidor/
AuthenticBrowserEdition_x64.CAB#Version=12,2,0,0" ')
    }
    else {
        return('CodeBase="http://su-servidor/
AuthenticBrowserEdition.CAB#Version=12,2,0,0" ')
    }
}

// Crear la instancia de objeto del complemento en función del
explorador que cargue la página
// -Firefox usa la etiqueta EMBED para incrustar complementos y permite
usar el atributo PLUGINSPAGE
// para redirigir a un archivo de instalación si el complemento
// no está instalado todavía;
// -IE usa la etiqueta <OBJECT> para incrustar complementos y permite
usar el atributo CODEBASE
// para señalar el archivo .cab necesario para la instalación
// si el complemento no está instalado todavía.

function createObject( codebase, clsid)
{
    if ( isFirefoxOnWindows() )
    {
        document.write ( '<embed ' +
'id="objPlugin" ' +
'type="application/x-authentic-scriptable-plugin" ' +
'width="100%" ' +
'height="60%" ' + 'PLUGINSPAGE="http://su-servidor/Authentic/
AuthenticFirefoxPlugin_trusted.xpi" ' +
'SaveButtonAutoEnable="true" ' +
'EntryHelpersEnabled="true" ' +
'LicServer="your-server" ' +
'LicCompany="Altova" ' +
'LicKey="XXXXXXXXXX" ' +
'XMLDataSaveUrl="http://su-servidor/Authentic/SaveFile_orgchart.xml">' +
'</embed> ' );
    }
}
```

```

        else if ( isIEOnWindows() )
    {
        document.write ( '<OBJECT ' +
            'id="objPlugIn" ' +
            'getCodeBase() ' +
            'Classid="clsid:B4628728-E3F0-44a2-BEC8-F838555AE780" ' +
            'width="100%" ' +
            'height="60%" ' +
            '>' +
            '<PARAM NAME="XMLDataSaveUrl" VALUE="http://su-servidor/Authentic/
SaveFile_OrgChart.xml"> ' +
            '<PARAM NAME="EntryHelpersEnabled" VALUE="TRUE"> ' +
            '<PARAM NAME="SaveButtonAutoEnable" VALUE="TRUE"> ' +
            '<PARAM NAME="LicServer" VALUE="su-servidor"> ' +
            '<PARAM NAME="LicCompany" VALUE="Altova"> ' +
            '<PARAM NAME="LicKey" VALUE="XXXXXXXXXX"> ' +
            '</OBJECT>');
    }
}

createObject();
// tras ejecutar createObject el objeto del complemento existe.
Inicializar la variable de javascript que se debe usar en la variables de los
scripts
var objPlugIn = document.getElementById('objPlugIn');
</script>

<br><br>
<button onclick="objPlugIn.StartEditing()"><span>Iniciar edición</
span></button>
<button onclick="objPlugIn.FindDialog()"><span>Buscar</span></button>
<button onclick="objPlugIn.ReplaceDialog();"><span>Reemplazar</span></
button>
<button onclick="BtnOnSave()"><span>Guardar</span></button>
<button onclick="alert ( objPlugIn.IsRowInsertEnabled );"><span>Probar</
span><br></button>
</center>

<script language="javascript" type="text/javascript">
    // suscripción a evento si se ejecuta en Firefox
    if ( isFirefoxOnWindows() )
    {
        objPlugIn.addEventListener("ControlInitialized",
InitAuthenticPluginPage, false);
    }
</script>

<script event="ControlInitialized" for="objPlugIn" language="javascript"
type="text/javascript">
    // suscripción a evento si se ejecuta en IE
    if ( isIEOnWindows() )
    {
        InitAuthenticPluginPage();
        //if ( isIE64OnWindows() ) alert("IE x64");
    }
</script>

```

```
</body>
</html>
```

Nota: el script anterior contiene la información sobre la licencia necesaria para activar Authentic Browser Enterprise Edition. Si faltan los tres parámetros LicServer, LicCompany y LicKey, entonces solamente se habilitarán las funciones de la edición Community Edition.

4 Paquetes de extensión para instalación por encargo

Si tiene pensado que la instalación se realice por encargo, los paquetes de extensión comprimidos de Authentic Browser (los archivos CAB, XPI y CRX) deben estar almacenados en el servidor. Cuando se acceda por primera vez a la [página HTML para el complemento de Authentic](#) desde el explorador cliente, las instrucciones de la página HTML provocan la descarga del paquete de instalación correspondiente desde el servidor al cliente, lo descomprime y lo instala en el cliente.

Los paquetes de extensión de Authentic Browser necesarios para la instalación por encargo se pueden descargar del sitio web de Altova. Después almacénelos en el servidor como archivos comprimidos (CAB, XPI o CRX, dependiendo del explorador que se utilice en el cliente para abrir la [página HTML para el complemento de Authentic](#)).

Internet Explorer 5.5 o superior (32 bits)	Archivo CAB (para IE de 32 bits)
Internet Explorer (64 bits)	Archivo CAB (para IE de 64 bits)
Firefox	Archivo XPI

Si un cliente puede usar varios exploradores o cualquier explorador (IE o Firefox), entonces debe almacenar todos los paquetes de extensión en el servidor. Si se desconoce el explorador cliente, incluya un script en la [página HTML para el complemento de Authentic](#) que detecte qué explorador se utiliza. El archivo correspondiente se descargará automáticamente desde el servidor al cliente. Para más información sobre este ejemplo de uso consulte el apartado [Página HTML para detectar el explorador](#).

En el servidor puede almacenar el archivo CAB para IE de 32 bits y el archivo CAB para IE de 64 bits. En la [página HTML para el complemento de Authentic](#) puede incluir un script que detecte si el explorador es de 32 o de 64 bits y que descargue el archivo CAB correspondiente. Esto se explica en el apartado [Página HTML para detectar el explorador](#).

Descargar y almacenar el archivo CAB/XPI/CRX

El archivo CAB, XPI o CRX se descarga del [sitio web de Altova](#) y se puede guardar en cualquier ubicación del servidor. Si quiere implementar la edición Authentic Browser Enterprise Edition, **debe almacenar el paquete de instalación en el servidor para el que se registró el producto**.

Los tres tipos de archivo de instalación (CAB, XPI y CRX) son formatos de archivo comprimidos. No descomprima estos archivos. La extracción e instalación en el cliente se lleva a cabo automáticamente cuando el cliente abre la [página HTML para el complemento de Authentic](#) por primera vez. La ubicación del archivo CAB, XPI o CRX en el servidor también se especifica en la [página HTML para el complemento de Authentic](#).

Altova Authentic 2015 Browser Edition

Configuración de clientes

Configuración de clientes

El equipo cliente es el equipo en el que se abre la [página HTML para el complemento de Authentic](#) y en el que se edita el documento XML. Para poder habilitar estas funciones el complemento Authentic Browser tiene que estar instalado en el explorador cliente que se usará para abrir la [página HTML para el complemento de Authentic](#).

El complemento Authentic Browser se puede instalar en el explorador cliente [automáticamente desde el servidor](#) cuando la [página HTML para el complemento de Authentic](#) se abre por primera vez en el explorador cliente. También puede instalarse [a mano](#) o con el [archivo de instalación MSI](#).

En esta sección explicamos cómo configurar el equipo cliente para trabajar con el complemento Authentic Browser, en concreto:

- Los [requisitos del explorador web](#) para poder habilitar las funciones de Authentic Browser.
- Los [métodos de instalación del complemento Authentic Browser](#) en los exploradores compatibles en el equipo cliente. También describe cómo [actualizar](#) y [desinstalar](#) Authentic Browser.
- [Configuración de seguridad de IE9](#) para garantizar el correcto funcionamiento del complemento.
- [Configuración de seguridad de IE10](#) para garantizar el correcto funcionamiento del complemento. Recuerde que Authentic Browser funcione en Internet Explorer 10, debe habilitar el modo compatible de IE 10.

1 Requisitos

Para que la página XML de Authentic se reproduzca correctamente, el equipo cliente debe tener:

- **Microsoft Internet Explorer 5.5 o superior o Mozilla Firefox (32 bits)** (preferiblemente la versión más reciente). El explorador abre la página HTML para Authentic y lleva a cabo las funciones codificadas en la página. El complemento Authentic Browser es compatible con IE (32 y 64 bits) y Firefox (32 bits).
- **Microsoft Internet Explorer 5.5 o superior**
Internet Explorer debe estar instalado porque la interfaz de la vista Authentic (que se abre en la ventana del explorador) se genera con Internet Explorer.

2 Complemento para exploradores Authentic Browser

Los siguientes apartados describen cómo instalar Authentic Browser de varias maneras y cómo actualizar y desinstalar el complemento.

- [Instalación por encargo](#)
- [Instalación manual con los paquetes de extensión](#)
- [Instalación manual con el archivo de instalación MSI](#)
- [Instalación de inserción con el archivo de instalación MSI](#)
- [Actualizaciones automáticas](#)
- [Desinstalación y deshabilitación](#)

Instalar varias versiones

Hay varias versiones de Authentic Browser (español, inglés, alemán, japonés) y cada una de ellas tiene versiones seguras y no seguras para cada explorador compatible (Microsoft Internet Explorer (32 y 64 bits) y Mozilla Firefox).

Si quiere puede instalar varias versiones del complemento en el mismo equipo cliente. Cada versión se presenta como un complemento diferente en el administrador de complementos del explorador.

Para más información consulte el apartado [Versiones de Authentic Browser](#).

2.1 Instalación por encargo

Lo mejor es instalar el complemento Authentic Browser automáticamente cuando lo necesite:

1. Los paquetes de extensión (archivos CAB, XPI o CRX) se almacenan en el servidor.
2. Cuando se abre la [página HTML para el complemento de Authentic](#) en el explorador del cliente, las instrucciones de la página HTML descargan el paquete de extensión correspondiente e instalan Authentic Browser en el explorador del cliente.

Para cada explorador web es necesario usar instrucciones diferentes en la página HTML.

Para Internet Explorer

Use el atributo CODEBASE del elemento OBJECT para especificar la dirección URL del archivo CAB.

Para más información consulte el [ejemplo para Internet Explorer](#).

Para Mozilla Firefox

Use el atributo pluginspage del elemento embed para especificar la dirección URL del archivo XPI. Para más información consulte el [ejemplo para Firefox](#).

Método alternativo para Mozilla Firefox

Añada un vínculo que apunte al paquete de extensión. Cuando se haga clic en este vínculo, se instalará el complemento. Por ejemplo: Haga clic aquí para instalar el complemento Authentic Browser.

Para más información sobre cómo configurar los paquetes de extensión en el lado servidor, consulte el apartado [Paquetes de extensión para instalación por encargo](#).

También puede consultar [este apartado](#) para conocer las diferentes versiones del complemento Authentic Browser.

2.2 Instalación manual con los paquetes de extensión

En algunos exploradores el complemento Authentic Browser también se puede instalar a mano.

Internet Explorer

En Internet Explorer no se pueden instalar complementos guardados en archivos CAB. En su lugar puede usar [el archivo de instalación MSI](#).

Firefox

Arrastre el archivo XPI hasta la ventana de Firefox. El administrador de complementos (**Herramientas | Complementos | Extensiones**) ofrece una lista de complementos instalados, donde debería aparecer Authentic Browser.

Para más información sobre las diferentes versiones de Authentic Browser, consulte [este apartado](#).

2.3 Instalación manual con el archivo de instalación MSI

El archivo de instalación MSI (*Microsoft Windows Installer*) para Authentic Browser se puede descargar del sitio web de Altova. Se trata de un archivo de instalación ejecutable (con una extensión .exe) que instala el complemento Authentic Browser en todos los exploradores web compatibles que estén instalados en el equipo cliente (Microsoft Internet Explorer de 32 y 64 bits y Mozilla Firefox).

Instalación

Descargue el archivo de instalación MSI en el equipo cliente y haga doble clic en él para iniciar la instalación. El complemento se instalará (en la carpeta C:\Archivos de programa (x86)\Altova\AuthenticBrowserPlugin...) y se integrará en los exploradores web compatibles que estén instalados. Si no quiere instalar el complemento en todos los exploradores, puede configurar la instalación en la pantalla *Instalación personalizada*.

Las versiones instaladas de Authentic Browser aparecen en la lista de programas *Agregar o quitar programas* del Panel de control del equipo cliente.

Versiones del archivo MSI

Para cada una de las distintas versiones de Authentic Browser hay un archivo de instalación MSI distinto. Por ejemplo, el archivo MSI para el complemento en inglés versión Trusted para complementos de 32 bits es diferente del archivo MSI en inglés versión Untrusted para exploradores de 32 bits.

Consulte [este apartado](#) para conocer todas las versiones del complemento Authentic Browser.

2.4 Instalación de inserción con el archivo de instalación MSI

Si la instalación de software nuevo la gestiona el equipo de soporte técnico y los archivos de instalación se distribuyen dentro de la red de administración, es conveniente usar un archivo de instalación MSI nativo. Sobre todo en el caso de Internet Explorer porque para instalar complementos en este explorador hacen falta derechos de administrador.

Desde el sitio web de Altova puede descargar el archivo de instalación MSI para el complemento Authentic Browser. Se trata de un archivo de instalación ejecuta (con la extensión .exe) que instala el complemento Authentic Browser en todos los exploradores compatibles que estén instalados en el equipo cliente (Microsoft Internet Explorer de 32 y 64 bits y Mozilla Firefox).

Instalación

Descargue el archivo de instalación MSI en el servidor de administración e inclúyalo en los procesos administrativos para instalar programas nuevos en equipos cliente. El complemento se instalará (en la carpeta C:\Archivos de programa (x86)\Altova\AuthenticBrowserPlugin\ ...) y se integrará en los exploradores web compatibles que estén instalados. Si no quiere instalar el complemento en todos los exploradores, puede configurar la instalación en la pantalla *Instalación personalizada*.

Para instalarlo en silencio, use el parámetro de entrada de la línea de comandos /q.

Las versiones instaladas de Authentic Browser aparecen en la lista de programas *Agregar o quitar programas* del Panel de control del equipo cliente.

Versiones del archivo MSI

Para cada una de las distintas versiones de Authentic Browser hay un archivo de instalación MSI distinto. Por ejemplo, el archivo MSI para el complemento en inglés versión Trusted para complementos de 32 bits es diferente del archivo MSI en inglés versión Untrusted para exploradores de 32 bits.

Consulte [este apartado](#) para conocer todas las versiones del complemento Authentic Browser.

2.5 Actualizaciones automáticas

La actualización del componente Authentic Browser se lleva a cabo de diferente manera en cada explorador web.

Internet Explorer

El valor del atributo `codebase` del elemento `object` de la [página HTML para el complemento de Authentic](#) puede llevar el sufijo `#Version=...` para indicar el número de versión del complemento que está disponible en el servidor. Si la versión del complemento que está instalado en el equipo cliente es inferior a la versión del atributo, se le preguntará al usuario si quiere actualizar el complemento.

Firefox

En Firefox las extensiones se actualizan de esta manera: el archivo `install.rdf` de la extensión de Firefox contiene un vínculo a un archivo `update.rdf` que está en un servidor de descargas. El archivo `install.rdf` especifica el número de versión del complemento que está instalado en el equipo del usuario. El archivo `update.rdf` enumera las versiones y los archivos de extensión que se pueden descargar del servidor. Si hay versiones más recientes que la instalada, se le informa de ello al usuario.

Este mecanismo de actualización no es compatible con Authentic Browser.

2.6 Desinstalación y deshabilitación

Si usó un archivo de instalación MSI para la instalación, en la lista *Agregar o quitar programas* del Panel de control debe aparecer el complemento Authentic Browser. Si lo desinstala desde esta lista, el complemento se quita del explorador y de la lista de programas.

Los complementos instalados en el explorador web aparecen como *habilitado* en el administrador de complementos del explorador. Si quita el complemento desde este administrador, el complemento aparecerá como *deshabilitado* pero seguirá en el disco del equipo cliente.

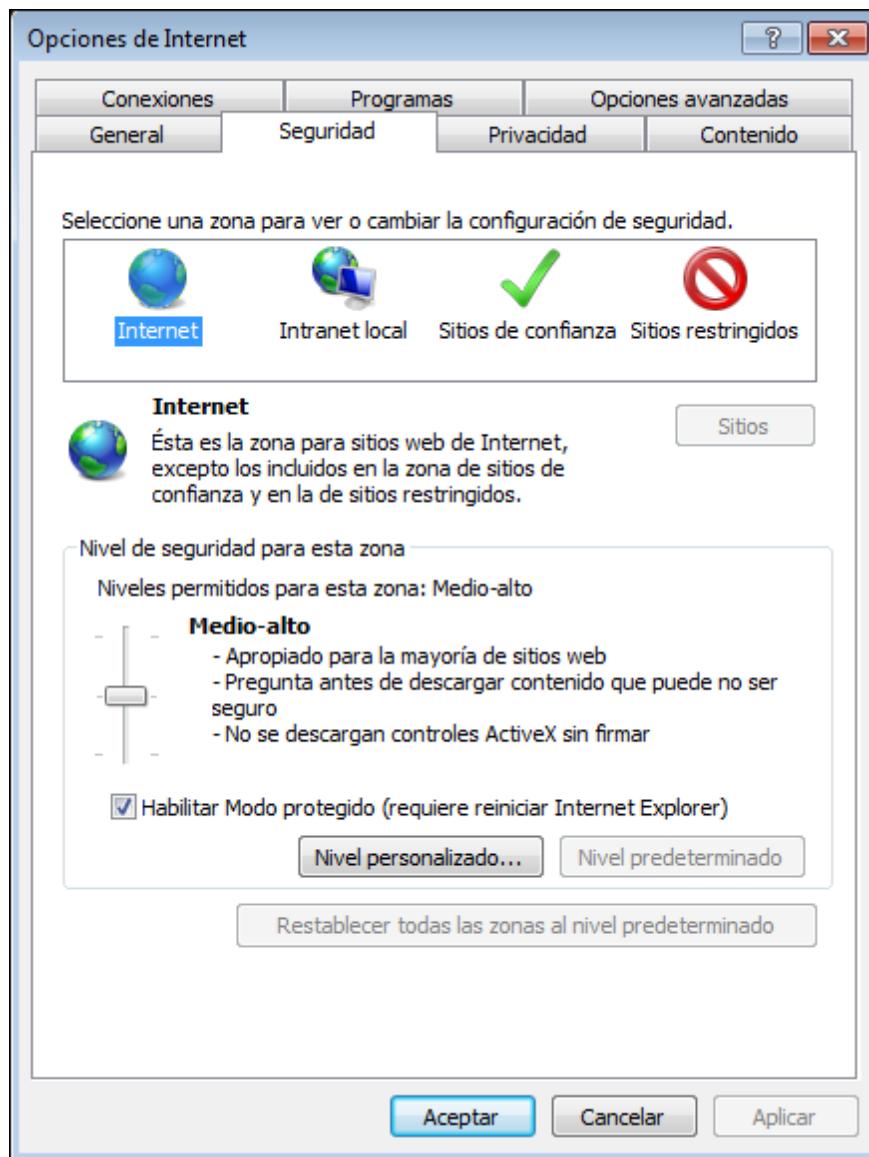
3 Configuración de seguridad en IE9

Si usa la [versión no segura de Authentic Browser](#) en Internet Explorer 9, puede que reciba este mensaje de error:

Internet Explorer bloqueó un control ActiveX. Es posible que esta página no se muestre correctamente.

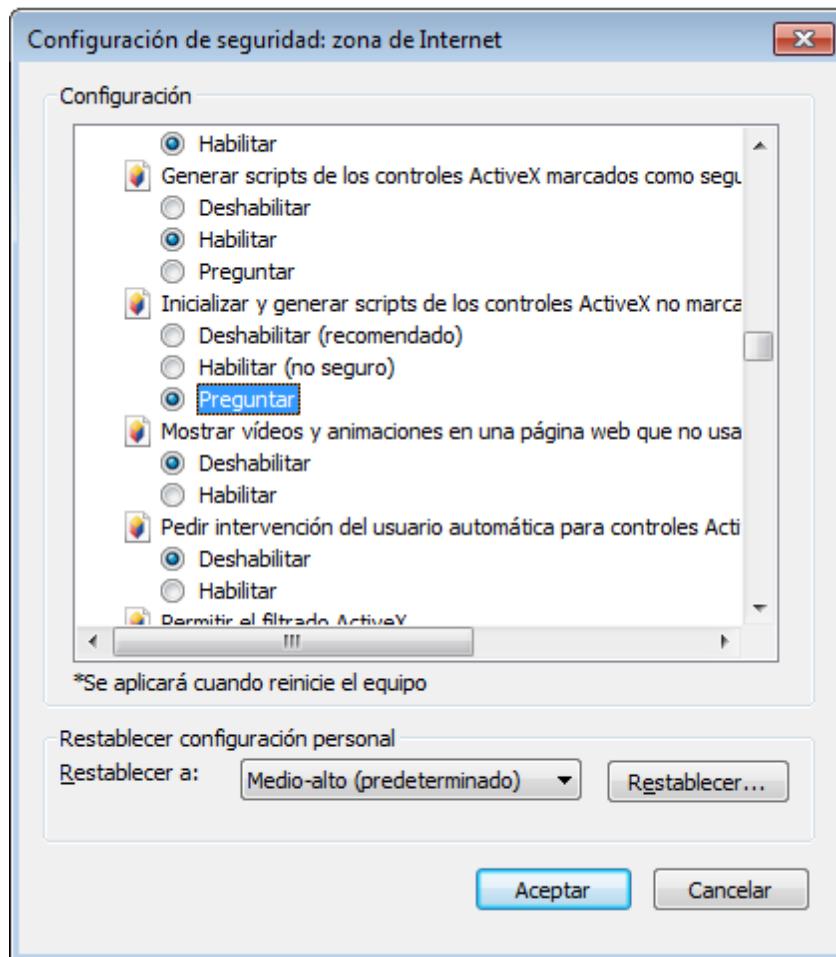
Para habilitar los controles ActiveX, debe configurar la opción de seguridad correspondiente en Internet Explorer, siguiendo las instrucciones que aparecen a continuación:

1. Abra el cuadro de diálogo "Opciones de Internet" (*imagen siguiente*) de Internet Explorer con el comando de menú **Herramientas | Opciones de Internet**.



2. Seleccione la pestaña *Seguridad* (*imagen anterior*) y la zona para la que quiere cambiar la configuración (Internet o intranet local).
3. Ahora haga clic en el botón **Nivel personalizado...** para abrir el cuadro de diálogo

"Configuración de seguridad: zona de Internet" (*imagen siguiente*).



4. Desplácese hasta ver la sección *Controles y complementos ActiveX* y hasta la opción *Inicializar y generar scripts de los controles ActiveX no marcados como seguros para scripts*. Seleccione el botón de opción *Preguntar* (*imagen anterior*). Para confirmar haga clic en **Aceptar**.
5. Reinicie Internet Explorer para aplicar la nueva configuración.

A partir de ahora cada vez que una [página HTML para el complemento de Authentic](#) intente cargar un archivo ActiveX, Internet Explorer le preguntará si quiere cargar o no el control.

4 Configuración de seguridad en IE10

Internet Explorer 10 (IE10) tiene dos modos:

- El modo estilo Metro, que ejecuta el modo de seguridad *Enhanced Protection Mode* y
- Y el modo escritorio, que ejecuta el modo de seguridad *Modo compatible*.

Para poder ejecutar el complemento Authentic Browser en IE10 debe usar el Modo compatible (**Herramientas | Opciones | Seguridad**).

Altova Authentic 2015 Browser Edition

Referencia

Referencia

En la referencia se enumeran y describen todos los mecanismos, objetos y enumeraciones de Authentic Browser.

- [Mecanismos](#)
- [Objetos](#)
- [Enumeraciones](#)

1 Mecanismos

Esta sección describe los mecanismos de Authentic Browser más frecuentes.

1.1 Eventos: punto de conexión (para IE)

Authentic Browser ofrece varios eventos de punto de conexión (consulte [Eventos: referencia](#)), para los que puede suministrar controladores de eventos (en los bloques SCRIPT de la página HTML).

Nota: la descripción de esta sección **corresponde a Internet Explorer solamente**.

Estos ejemplos incluyen bloques SCRIPT para los eventos ControlInitialized y SelectionChanged:

ControlInitialized

Este evento de punto de conexión se desencadena inmediatamente después de crearse e inicializarse el control. Fuera del controlador del evento ControlInitialized se pueden controlar más scripts de inicio para el control.

```
<SCRIPT LANGUAGE=javascript FOR=objPlugIn EVENT=controlinitialized>
    // añada aquí su código
</SCRIPT>
```

SelectionChanged

El evento SelectionChanged se desencadena cada vez que cambia al selección actual en la vista. Puede usar un bloque SCRIPT para ejecutar el código personal del evento.

```
<SCRIPT LANGUAGE=javascript FOR=objPlugIn EVENT=selectionchanged>
    // añada aquí su código
</SCRIPT>
```

Recuerde que el objeto [Authentic.event](#) no se rellena cuando ocurre este evento. Si en su script hay controladores de eventos registrados, las propiedades del objeto [Authentic.event](#) contienen valores del último evento que ocurrió.

El objeto [Authentic.CurrentSelection](#) ahora contiene información válida.

Cargar archivos SPS, XSD y XML sin la intervención del usuario

Si quiere cargar archivos .sps, .xsd y .xml sin la intervención del usuario cuando se cargue la página HTML, lo mejor es escribir un controlador de eventos que controle el evento ControlInitialized. Otra opción es usar un contenedor de propiedades, como en el segundo ejemplo que aparece a continuación:

Método recomendado:

```
<SCRIPT LANGUAGE="javascript" FOR=objPlugIn EVENT="ControlInitialized">
objPlugIn.SchemaLoadObject.URL = "http://suservidor/OrgChart.xsd"
objPlugIn.XMLDataLoadObject.URL = "http://suservidor/OrgChart.xml"
objPlugIn.DesignDataLoadObject.URL = "http://suservidor/OrgChart.sps"
objPlugIn.StartEditing()
</SCRIPT>
```

Método alternativo:

```
<OBJECT id=objPlugIn style="WIDTH:600px; HEIGHT:500px"
codeBase="http://suservidor/cabfiles/
AuthenticBrowserEdition.CAB#Version=12,3,0,0"
classid=clsid:B4628728-E3F0-44a2-BEC8-F838555AE780>
<PARAM NAME="XMLDataURL" VALUE="http://suservidor/OrgChart.xml">
<PARAM NAME="SPSDataURL" VALUE="http://suservidor/OrgChart.sps">
<PARAM NAME="SchemaDataURL" VALUE="http://suservidor/OrgChart.xsd">
</OBJECT>
```

No recomendamos cargar estos archivos en un controlador de eventos que controle los elementos "body" del evento "onload" porque puede que el control de Authentic Browser se inicialice después de que se desencadene el evento "onload". Si eso ocurre, los métodos y las propiedades del complemento no estarán disponibles y los archivos no se cargarán. Consulte el apartado [El elemento OBJECT](#) para obtener más información.

Método no recomendado:

```
<SCRIPT LANGUAGE="javascript">
function load () {
    objPlugIn.SchemaLoadObject.URL = "http://suservidor/OrgChart.xsd"
    objPlugIn.XMLDataLoadObject.URL = "http://suservidor/OrgChart.xml"
    objPlugIn.DesignDataLoadObject.URL = "http://suservidor/OrgChart.sps"
    objPlugIn.StartEditing()
}
</SCRIPT>

<body onload = "load files">
```

1.2 Eventos: añadir agentes de escucha de eventos (para Firefox)

Por cada evento de punto de conexión dados por Authentic Browser, puede escribir controladores de eventos en los bloques SCRIPT de la página HTML con ayuda del método addEventListener.

Nota: la descripción de esta sección **corresponde a Firefox solamente**.

Este ejemplo muestra cómo suministrar un controlador de eventos para el evento SelectionChanged en Firefox:

SelectionChanged

El evento SelectionChanged se desencadena cada vez que la selección actual cambia en la vista. Puede usar un bloque SCRIPT para ejecutar el código personal del evento.

```
<SCRIPT LANGUAGE="javascript">
var selCount = 0;
function OnSelectionChanged()
{
    selCount = selCount + 1;
    selectionCounter.value = "SelectionCount = " + selCount;
}
var objPlugIn = document.getElementById('objPlugIn');
objPlugIn.addEventListener("selectionchanged", OnSelectionChanged, false)
</SCRIPT>
```

Recuerde que el objeto [Authentic.event](#) no se rellena cuando ocurre este evento. Si en su script hay controladores de eventos registrados, las propiedades del objeto [Authentic.event](#) contienen valores del último evento que ocurrió.

El objeto [Authentic.CurrentSelection](#) ahora contiene información válida.

1.3 Eventos: botones de las barras de herramientas

Cada botón de las barras de herramientas tiene un comportamiento predeterminado que puede cambiarse. El evento `AuthenticCommand` sirve para añadir tareas adicionales o para redefinir por completo el comportamiento predeterminado de los botones de las barras de herramientas. Los scripts pueden usar el evento `AuthenticCommand` para recibir una notificación cada vez que el usuario haga clic en un ícono de una barra de herramientas.

Recuerde que no todos los comandos (de la colección `Authentic.UICommands`) tiene su propio evento asociado. Para averiguar en qué ícono hizo clic el usuario, el script debe revisar la propiedad `AuthenticEvent.srcElement`, que contiene una referencia a un objeto `AuthenticCommand`.

Ejemplo

```
// controlador de eventos para OnDocEditCommand
<SCRIPT LANGUAGE=javascript FOR=objPlugIn EVENT=doceditcommand>
    // nos interesa el botón k_CommandSave
    if(objPlugIn.event.srcElement.CommandID == 1)
    {
        // en lugar de HTTP PUT
        // queremos usar HTTP POST
        objPlugIn.SavePOST();

        // no continúa una ejecución estándar
        objPlugIn.event.cancelBubble = true;
    }
</SCRIPT>
```

Referencia

Consulte el apartado [AuthenticToolbarButton.CommandID](#) para ver los comandos disponibles.

1.4 Eventos: referencia

Lista de eventos de punto de conexión

Nombre	desde TypeLib	Descripción
SelectionChanged	1.0	Cambió la selección en el editor. El objeto Authentic.CurrentSelection ahora contiene información válida. Las propiedades del objeto Authentic.event no están definidas.
ControllInitialized	1.2	El control está totalmente cargado e inicializado. Las propiedades del objeto Authentic.event no están definidas.
dragover	1.8	Está produciéndose una operación de arrastrar.
drop	1.8	Está produciéndose una operación de colocar.
keydown	1.8	Se pulsó una tecla pero no se soltó.
keyup	1.8	Se dejó de pulsar una tecla del teclado. Es el evento que se suele usar para reaccionar a pulsaciones de teclas.
keypressed	1.8	Con cualquier pulsación de teclado.
mousemove	1.8	Se movió el puntero del mouse.
buttonup	1.8	Se dejó de pulsar uno de los botones del mouse.
buttondown	1.8	Se pulsó uno de los botones del mouse.
contextmenu	1.8	Se envía después de recibir WM_CONTEXTMENU.
editpaste	1.8	Se le llama antes de que tenga lugar una operación de pegar.
editcut	1.8	Se le llama antes de que tenga lugar una operación de cortar.
editcopy	1.8	Se le llama antes de que tenga lugar una operación de copiar.
editclear	1.8	Se le llama antes de que tenga lugar una operación de borrar.
doceditcommand	1.8	Cuando se ejecuta un comando de Authentic y para implementar un controlador de comandos personal. Consulte Eventos: Toolbarbuttons . Las propiedades del objeto Authentic.event no están definidas.
butondoubleclick	1.8	Se pulsó dos veces seguidas uno de los botones del mouse.

Si no se menciona ninguna excepción en la descripción, las propiedades del objeto [Authentic.event](#) se definen al llamar al controlador de eventos.

Nota: estos controladores de eventos son sincrónicos, es decir, se les llama en cuanto ocurre el evento.

1.5 Acceder al contenido del documento y modificarlo

Para acceder al contenido del documento y modificarlo puede usar los objetos `AuthenticRange`, `AuthenticView` y `Authentic` (así como sus propiedades y métodos).

Cuando haya una coincidencia entre las funciones de las interfaces `AuthenticRange` y `AuthenticView`, por una parte, y la interfaz que aporta el objeto `Authentic`, por otra, la interfaz preferida es `AuthenticRange` y `AuthenticView`. Las funciones coincidentes del objeto `Authentic` se están eliminando de forma gradual y serán obsoletas en futuras versiones.

1.6 Operaciones de edición

Cuando los datos XML se presentan en la vista Authentic, los elementos se pueden manipular uno por uno mediante operaciones de edición estándar (cortar, copiar, pegar, etc.). Sin embargo, no todos los elementos de datos XML se pueden editar. Por tanto, primero hace falta probar si está permitido editarlos. El mecanismo utilizado es este: primero compruebe si la operación de edición está habilitada. Si lo está, llame al método que debe realizar la operación de edición. El único método que no tiene un test de prueba es el método `EditSelectAll`, que selecciona automáticamente todos los elementos que aparecen en el documento.

A continuación aparecen las propiedades y los métodos que realizan operaciones de edición. Cada propiedad devuelve un valor booleano. Los métodos no tienen parámetros.

IsEditUndoEnabled	Authentic.EditUndo	Deshacer una operación de edición.
IsEditRedoEnabled	Authentic.EditRedo	Rehacer una operación de edición.
IsEditCopyEnabled	Authentic.EditCopy	Copiar el texto seleccionado en el portapapeles.
IsEditCutEnabled	Authentic.EditCut	Cortar el texto seleccionado y copiarlo en el portapapeles.
IsEditPasteEnabled	Authentic.EditPaste	Pegar el texto del portapapeles en la posición actual del cursor.
IsEditClearEnabled	Authentic.EditClear	Borrar el texto seleccionado del documento XML.

1.7 Buscar y reemplazar

El método [Authentic.FindDialog](#) abre el cuadro de diálogo "Buscar", donde el usuario puede insertar el término de búsqueda.

El método [Authentic.FindNext](#) encuentra la siguiente instancia del término de búsqueda. El método `FindNext` se puede probar con la propiedad booleana `IsFindNextEnabled`.

Una variación del método `FindDialog` es el método [Authentic.ReplaceDialog](#). Esta operación busca un término concreto y lo reemplaza con un valor indicado por el usuario en el cuadro de diálogo "Reemplazar". Si después se llama al método `FindNext`, se busca y reemplaza la siguiente instancia del término de búsqueda.

1.8 Operaciones en filas de tablas

En la vista Authentic se pueden crear tablas dinámicas a partir de estructuras de elementos repetidos. En la tabla dinámica cada fila representa una instancia del elemento repetido. Una vez creada la tabla dinámica, el usuario de la vista Authentic puede manipular las filas y sus datos. Estas operaciones en las filas se pueden llevar a cabo con ayuda de un script.

Si quiere usar un script externo para realizar operaciones en filas, debe seguir estos dos pasos:

1. El primer paso consiste en comprobar si la fila en la que está el cursor usa una propiedad. Se usa una propiedad como `IsRowInsertEnabled`, por ejemplo, que devuelve el valor True o False.
2. Si el valor devuelto es True, se puede llamar al método de fila necesario.

A continuación aparecen las propiedades y los métodos que realizan operaciones en filas. Cada propiedad devuelve un valor booleano. Los métodos no tienen parámetros.

<code>IsRowInsertEnabled</code>	Authentic.RowInsert	Operación de inserción de filas.
<code>IsRowAppendEnabled</code>	Authentic.RowAppend	Operación de anexión de filas.
<code>IsRowDeleteEnabled</code>	Authentic.RowDelete	Operación de eliminación de filas.
<code>IsRowMoveUpEnabled</code>	Authentic.RowMoveUp	Subir los datos XML una fila.
<code>IsRowMoveDownEnabled</code>	Authentic.RowMoveDown	Bajar los datos XML una fila.
<code>d</code>		
<code>IsRowDuplicateEnabled</code>	Authentic.RowDuplicate	Hacer una copia de la fila actual.

1.9 Teclas de acceso rápido

Estas son las teclas de acceso rápido que puede usar dentro de la ventana de Authentic Browser:

CTRL + P	Imprimir documento
CTRL + Z	Deshacer
CTRL + Y	Rehacer
CTRL + X	Cortar
CTRL + C	Copiar
CTRL + V	Pegar
CTRL + A	Seleccionar todo
CTRL + F	Abrir el cuadro de diálogo "Buscar"
CTRL + H	Abrir el cuadro de diálogo "Buscar y reemplazar"

1.10 Botones de estado del texto

La barra de herramientas de la vista Authentic del complemento Authentic Browser ofrece iconos de estado del texto. Estos iconos insertan elementos que tienen propiedades de formato de texto predefinidas. Los requisitos para poder usar esta función son:

- el elemento que se debe crear como ícono de estado del texto debe estar declarado como plantilla global en el esquema
- y el archivo SPS debe contener las definiciones necesarias (consulte la documentación de Altova StyleVision).

El complemento necesita tener la ruta de acceso del archivo .bmp que se usará como ícono de estado del texto o como botón en la barra de herramientas. Esta URL del archivo de imagen del ícono se suministra como valor de [Authentic.TextStateBmpURL](#).

1.11 Ayudantes de entrada

El complemento Authentic Browser permite usar ventanas de ayudantes de entrada en la vista Authentic. De este modo, el usuario de la vista Authentic tiene acceso a los elementos, atributos y entidades que están permitidos en cada posición del documento. Los ayudantes de entrada están deshabilitados por defecto y no aparecen a no ser que se activen explícitamente. Para habilitar los ayudantes de entrada puede utilizar estos parámetros en la etiqueta OBJECT ([Internet Explorer](#)) o como atributo del elemento EMBED ([Firefox](#)):

EntryHelpersEnabled	TRUE / FALSE
EntryHelperSize	Tamaño en pixels
EntryHelperAlignment	Toma los valores de SPYAuthenticToolbarAlignment
EntryHelperWindows	Toma los valores de SPYAuthenticEntryHelperWindows . Se permiten todas las combinaciones posibles

En lugar de usar los parámetros en las etiquetas OBJECT o EMBED también puede usar las propiedades y métodos en la API:

Propiedades

- [Authentic.EntryHelpersEnabled](#)
- [Authentic.EntryHelperAlignment](#)
- [Authentic.EntryHelperSize](#)
- [Authentic.EntryHelperWindows](#)

Método

- [Authentic.RedrawEntryHelpers](#)

1.12 Paquetes

El complemento Authentic Browser admite el uso de paquetes para expandir las funciones del módulo del programa.

Los paquetes se almacenan en el servidor y se instalan en el cliente por encargo desde la interfaz gráfica de Authentic Browser directamente. Una vez instalado, el paquete se activa cada vez que se inicia sesión y hasta que el usuario lo quite.

Actualmente Authentic Browser es compatible con paquetes de corrector ortográfico.

Esta sección tiene dos apartados:

- [Trabajar con paquetes](#), que describe el funcionamiento de los paquetes.
- [Paquetes del corrector ortográfico](#), que explica cómo se almacenan en el servidor los paquetes del corrector ortográfico, cómo se instalan y cómo se utilizan.

1.12.1 Trabajar con paquetes

Los paquetes se almacenan en el servidor y se instalan localmente (en el cliente) por encargo desde la interfaz gráfica de Authentic Browser directamente. Una vez instalado, el paquete se activa cada vez que se inicia sesión y hasta que el usuario lo quite.

Gestión de paquetes

Authentic Browser ofrece una función que permite al usuario controlar los paquetes disponibles. Esta función está disponible en el cuadro de diálogo "Gestión de paquetes".

Para habilitar la función de gestión de paquetes debe usar el parámetro `LoaderSettingsFileURL` en la [página HTML para el complemento de Authentic](#). Este parámetro especifica la URL del archivo `LoaderSettings` para la gestión de paquetes (que puede residir en cualquier ubicación a la que se tenga acceso):

- En una [página HTML para Internet Explorer](#) el parámetro `LoaderSettingsFileURL` se usa como elemento `PARAM` secundario del elemento `OBJECT`:

```
<OBJECT>
  ...
  <PARAM NAME="LoaderSettingsFileURL" VALUE="http://www.server.com/
AuthenticFiles/XMLSpyPlugInLoaderSettings.xml" />
</OBJECT>
```

- En una [página HTML para Firefox](#) el parámetro `LoaderSettingsFileURL` se usa como atributo del elemento `EMBED`:

```
<EMBED
  ...
  LoaderSettingsFileURL="http://www.server.com/AuthenticFiles/
XMLSpyPlugInLoaderSettings.xml" />
```

El archivo `LoaderSettings` contiene las definiciones de los paquetes y sus URL. Más abajo puede ver un fragmento de un archivo `LoaderSettings`.

Ejemplo de archivo XML LoaderSettings

El elemento de documento del archivo XML `LoaderSettings` es `loadersettings` (*ver fragmento más abajo*). Este elemento puede contener varios secundarios `package` y varios elementos `zippackage`.

- El elemento `package` hace referencia a los paquetes de ortografía `.pck` antiguos y se usa en las versiones de Authentic Browser anteriores a la versión 2011r3. Estas versiones de Authentic Browser ignoran el elemento `zippackage`.
- El elemento `zippackage` hace referencia a los paquetes de ortografía `.zip` y se usa en las versiones de Authentic Browser posteriores a la versión 2011r3. Estas versiones de Authentic Browser ignoran el elemento `package`.

Para obtener más información consulte el apartado [Paquetes del corrector ortográfico](#).

```
<?xml version="1.0" encoding="UTF-8"?>
<loadersettings>

<zippackage mode="user_demand" category="spelling">
```

```
<packageurl>Portuguese (Brazilian).zip</packageurl>
<description>Portuguese (BR) language pack.</description>
</zippackage>

<zippackage mode="user_demand" category="spelling">
<packageurl>Portuguese (Brazilian).zip</packageurl>
<description>Portuguese (BR) language pack.</description>
</zippackage>

<package mode="user_demand" id="SentrySpellChecker_EAM_only"
category="spelling" version="1">
<packageurl>PlugIn/SentrySpellChecker_EAM_only.pck</packageurl>
<description>Sentry Spellchecker (EN-US)</description>
</package>

<package mode="user_demand" id="SentrySpellChecker_EALL"
category="spelling" version="1">
<packageurl>PlugIn/SentrySpellChecker_EALL.pck</packageurl>
<description>Sentry SpellChecker EN with Legal and Medical.</description>
</package>

</loadersettings>
```

Es necesario tener en cuenta que:

- La ubicación del paquete se debe especificar como contenido del secundario **packageurl** del elemento **package** o **zippackage**. La ruta de acceso puede ser absoluta o relativa al archivo HTML que llama al archivo **LoaderSettings**. El paquete puede estar en cualquier ubicación del servidor.
- Si se elimina un paquete del archivo XML (eliminando el elemento **package**) el paquete no se podrá instalar en el cliente.
- El atributo **mode** del elemento **package** especifica el nivel de influencia que puede tener el usuario sobre la decisión de instalar el paquete. Se pueden usar estos valores:
 1. **user**: se le pregunta al usuario cada vez que se inicia Authentic Browser si desea instalar el paquete.
 2. **user_demand**: instala el paquete cuando lo solicita el usuario. El usuario lo solicita haciendo clic en el botón de la barra de herramientas para revisar la ortografía o para abrir el cuadro de diálogo "Gestión de paquetes".
 3. **force**: provoca la instalación del paquete sin notificar al usuario.
- El contenido del elemento secundario **description** del elemento **package** se puede editar y se usará como texto descriptivo en el cuadro de diálogo "Gestión de paquetes".

Una sola instalación

Si actualiza su versión de Authentic Browser e instala una versión más reciente, no hace falta volver a instalar los paquetes que ya están instalados en el cliente. Los paquetes que ya están instalados se utilizarán en la nueva versión. Authentic Browser lee el archivo **LoaderSettings** del

servidor, recupera la información del paquete de este archivo y comprueba si el paquete está instalado localmente.

Nota: los paquetes del corrector ortográfico compatibles con la versión 2011r3 y superior (paquetes .zip) usan un corrector ortográfico distinto al de las versiones anteriores (paquetes .pck). Las versiones 2011r3 y superiores de Authentic Browser usan paquetes .zip. Las versiones anteriores de Authentic Browser usan paquetes .pck.

1.12.2 Paquetes del corrector ortográfico

A partir de la versión 2011r3, Authentic Browser utiliza diccionarios Hunspell.

Ubicación de la instalación

En el equipo cliente los diccionarios están en la carpeta `Lexicons` situada en esta ubicación:

En Windows 7/8: `C:\ProgramData\Altova\SpellChecker\Lexicons`

En Windows XP: `C:\Documents and Settings\All Users\Application Data\Altova\SharedBetweenVersions\SpellChecker\Lexicons`

Si hay algún producto de Altova se instaló en el equipo cliente, entonces se instalaron también diccionarios integrados en la carpeta `Lexicons` de la ubicación correspondiente. Todos los diccionarios instalados se comparten con todos los usuarios del equipo y con los demás productos de Altova que estén instalados.

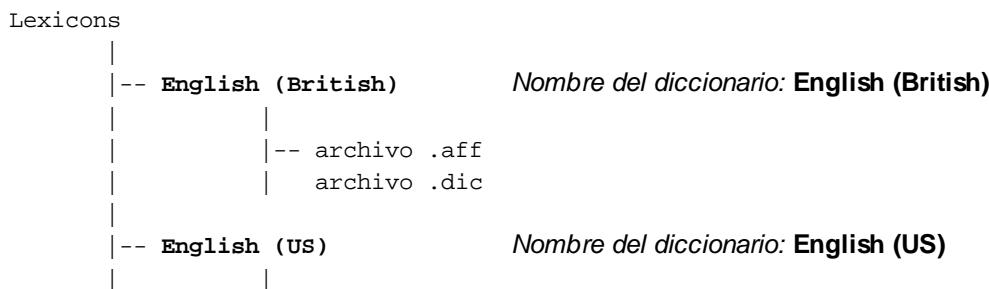
Puede descargar más diccionarios desde el [sitio web de Altova](#) o desde otros sitios web, como por ejemplo <http://wiki.services.openoffice.org/wiki/Dictionaries> o <http://extensions.services.openoffice.org/en/dictionaries>. Como los diccionarios Hunspell están basados en los diccionarios Myspell, estos también se pueden instalar en la carpeta `Lexicons`.

Nota: los diccionarios integrados en el software de Altova no refleja ningún tipo de preferencia por parte de Altova en cuanto al uso de ciertos idiomas. Simplemente se basa en la disponibilidad de diccionarios que pueden distribuirse con software comercial, como las licencias [MPL](#), [LGPL](#) o [BSD](#). Existen muchos otros diccionarios de código abierto pero se distribuyen con licencias más restrictivas, como la licencia [GPL](#). Muchos de estos diccionarios se pueden descargar desde el [sitio web de Altova](#). Recuerde que es decisión suya aceptar o no las condiciones de uso de la licencia del diccionario elegido. También es responsabilidad suya comprobar si el diccionario puede utilizarse en su equipo o no.

Archivos de los diccionarios y estructura de la carpeta de diccionarios en el equipo cliente

Cada diccionario del equipo cliente está compuesto por dos archivos de diccionario Hunspell: el archivo `.aff` y el archivo `.dic`.

Todos los diccionarios están instalados en la carpeta `Lexicons` de la ubicación correspondiente. Dentro de la carpeta `Lexicons` cada idioma se almacena en una carpeta distinta: `<idioma>\<archivos del diccionario>`. El nombre de la carpeta del idioma equivale al nombre que tiene el diccionario en Authentic Browser. A continuación puede ver la estructura de la carpeta `Lexicons` y el nombre correspondiente de los diccionarios tal y como aparecerían en Authentic Browser.



```
|           |-- archivo .aff  
|           |   archivo .dic  
  
|-- Spanish                         Nombre del diccionario: Spanish  
|           |  
|           |-- archivo .aff  
|           |   archivo .dic
```

Archivos de los diccionarios en el servidor

En el servidor los diccionarios se deben guardar en archivos ZIP. Cuando el archivo ZIP se descarga e instala en el cliente, se crea una carpeta del idioma en la carpeta `Lexicons` (ver más arriba) y los archivos `.aff` y `.dic` descomprimidos se colocan en esta carpeta. Por ejemplo, el archivo `English (British).zip` crea la carpeta: `<ruta de acceso>\Lexicons\English (British)` e instala los archivos `.aff` y `.dic` en esta carpeta.

También se pueden comprimir varios archivos ZIP en un solo archivo ZIP, que se puede guardar después en el servidor. Cuando este archivo ZIP se descarga e instala en el cliente, el nombre de los archivos ZIP del directorio más interior se usarán para el nombre de las carpetas de la carpeta `Lexicons`. Por ejemplo, los archivos `English (British).zip` y `English (US).zip` se pueden comprimir en un archivo ZIP llamado `English.zip`, que se coloca en el servidor. Una vez completada su instalación, se crearán dos carpetas en la carpeta `Lexicons`: `<ruta de acceso>\Lexicons\English (British)` y `<ruta de acceso>\Lexicons\English (US)`.

Cómo se instalan los diccionarios

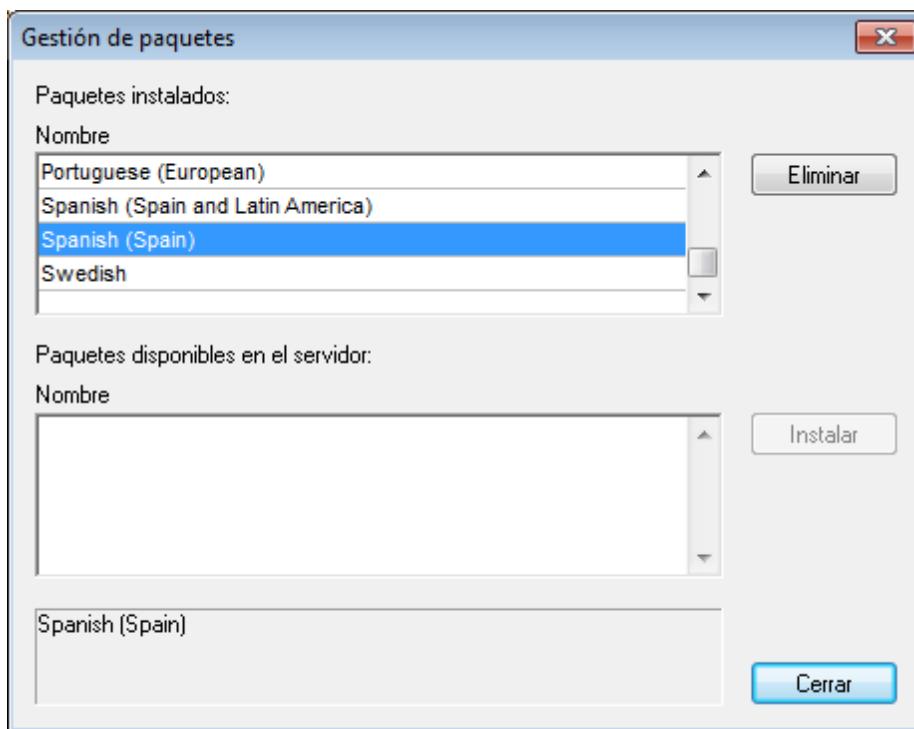
Cuando se abre una página HTML para el complemento Authentic Browser en el explorador, los paquetes de instalación se pueden instalar en el cliente bien automáticamente, bien cuando lo solicite el usuario. Así funciona el mecanismo de instalación:

1. La página HTML que se abre en Authentic Browser remite al [archivo LoaderSettings](#).
2. El [archivo LoaderSettings](#) remite al paquete del corrector ortográfico que está en el servidor. El [archivo LoaderSettings](#) también especifica, por cada paquete, si el paquete se debe instalar automáticamente justo después de que se cargue la página HTML o si se debe preguntar al usuario por la instalación.
3. El paquete del diccionario se instala en el cliente desde el archivo comprimido del servidor.

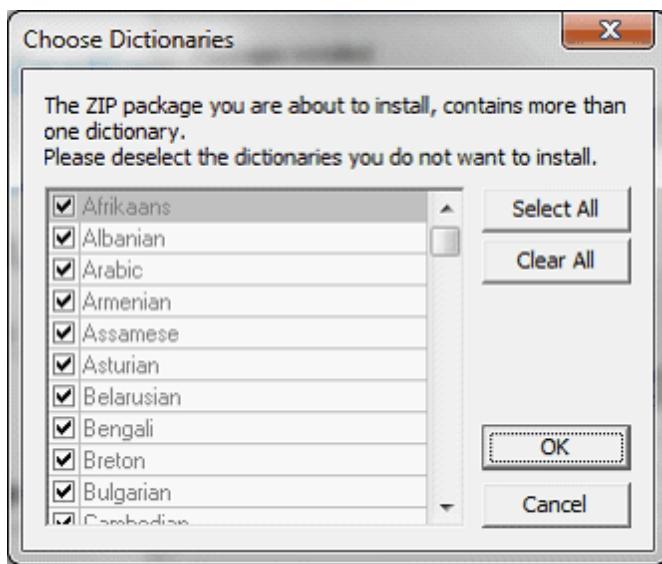
Instalación de los diccionarios

Los paquetes del corrector ortográfico se almacenan en el servidor como archivos comprimidos ZIP y se pueden instalar localmente desde la interfaz gráfica de Authentic Browser directamente. Para instalar un paquete haga clic en el ícono **Gestión de paquetes** de la barra de herramientas de Authentic Browser. Esto abre el cuadro de diálogo "Gestión de paquetes" (*Imagen siguiente*). Este cuadro de diálogo también aparece si no hay ningún diccionario instalado y se hace clic en el ícono **Corrección ortográfica**.

En el cuadro de diálogo "Gestión de paquetes" se enumeran todos los paquetes que están disponibles en el servidor. Los nombres que aparecen en el cuadro de diálogo equivalen a los nombres de los archivos ZIP. Seleccione qué archivos desea instalar y después haga clic en **Instalar**. Para eliminar paquetes instalados, seleccione un paquete del panel superior y haga clic en **Eliminar**.



Si el archivo ZIP seleccionado contiene varios paquetes ZIP, aparece el cuadro de diálogo "Elegir diccionarios" (imagen siguiente). En este cuadro de diálogo puede elegir qué paquetes desea instalar o desactivar la casilla de los paquetes que no desea instalar.



Si la instalación de un paquete sobrescribirá un paquete ya existente, aparece un mensaje de advertencia. El usuario de Authentic Browser puede continuar y sobrescribir el paquete actual o cancelar la instalación.

1.13 Interfaz XMLData

XMLData ofrece acceso a los elementos del archivo XML que está visible. Permite realizar todas las modificaciones necesarias a los elementos de la estructura XML. La principal función de XMLData es:

1. Acceder a los nombres y los valores de todo tipo de elementos (p. ej. elementos, atributos...).
2. Crear elementos nuevos de todo tipo.
3. Insertar y anexar elementos nuevos.
4. Borrar elementos secundarios ya existentes.

Si ya conoce la interfaz XMLData porque la usó en la API de XMLSpy, debe tener en cuenta algunos aspectos cuando se creen e inserten elementos nuevos en el archivo XML o cuando se cambie el nombre de los elementos ya existentes. Para más información siga leyendo este apartado.

Estructura de XMLData

Para poder usar la interfaz XMLData, antes es necesario conocer cómo se crean asignaciones de datos entre el archivo XML y la estructura XMLData. También es muy importante saber que XMLData no tiene una rama separada de objetos para los atributos.

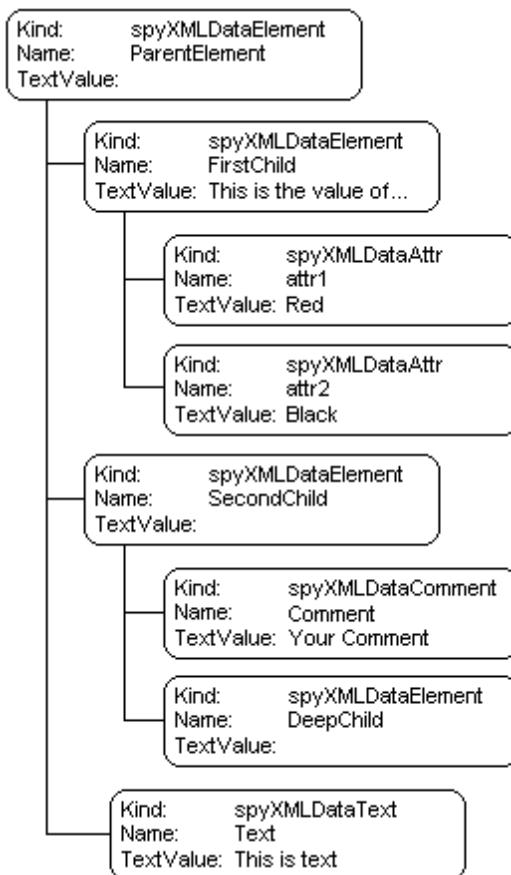
Los atributos de un elemento también son sus secundarios. La propiedad [XMLData.Kind](#) le ofrece la oportunidad de diferenciar entre tipos de secundarios distintos del mismo elemento.

Ejemplo

Este código XML,

```
<ElementoPrimario>
  <PrimerSecundario atr1="Rojo" atr2="Negro">
    Este es el valor de PrimerSecundario
  </PrimerSecundario>
  <SegundoSecundario>
    <!--Comentario-->
    </SegundoSecundario>
    Este es el texto
  </ElementoPrimario>
```

se asigna a esta estructura de objetos de XMLData:



El primario de todos los elementos XML que hay dentro de un archivo es la propiedad [Authentic.XMLRoot](#). Use este objeto XMLData para obtener referencias a los demás elementos XML de la estructura.

Nombre y valor de los elementos

Para conocer y modificar el nombre y el valor de todo tipo de elementos XML use las propiedades [XMLData.Name](#) y [XMLData.TextValue](#). Es posible que varios tipos de objetos de XMLData y elementos vacíos no tengan asociado ningún valor de texto.

No se recomienda cambiar el nombre de un elemento XML ya existente en la vista Authentic. El nombre tiene una gran relevancia en la forma en que la vista Authentic presenta el contenido del elemento. Consulte la documentación de Altova StyleVision para obtener más información.

Creación e inserción de objetos XMLData nuevos

Estas son las instrucciones para crear una entidad XML nueva:

1. Cree el objeto XMLData nuevo con ayuda del método [Authentic.CreateChild](#). Defina su nombre y su valor antes de insertar la entidad XML nueva (ver punto 3 más abajo).
2. Encuentre la ubicación correcta del objeto XMLData nuevo.
Para insertar un objeto XMLData nuevo primero debe obtener una referencia al primario. Si el secundario nuevo debe ser el último secundario del primario, utilice el método

[XMLData.AppendChild](#) para insertar el objeto XMLData. Si el secundario nuevo debe colocarse en otra posición de la secuencia de objetos secundarios, entonces puede usar [XMLData.GetFirstChild](#) y [XMLData.GetNextChild](#) para mover el iterador al secundario antes del cual debe insertarse el secundario nuevo.

3. Inserte el secundario nuevo con [XMLData.InsertChild](#). El secundario nuevo se inserta inmediatamente antes del secundario actual. En la vista Authentic puede ocurrir que junto con el elemento recién creado se añadan elementos secundarios adicionales al archivo XML. Esto depende de la configuración del nodo, que se puede modificar en StyleVision.

En el ejemplo siguiente se añade un tercer secundario entre los elementos <PrimerSecundario> y <SegundoSecundario>:

```
Dim objParent
Dim objChild
Dim objNewChild

Set objNewChild = objPlugIn.CreateChild(spyXMLElement)
objNewChild.Name = "OneAndAHalf"

'objParent is set to <ElementoPrimario>
'GetFirstChild(-1) gets all children of the parent element
'and move to <SegundoSecundario>
Set objChild = objParent.GetFirstChild(-1)
Set objChild = objParent.GetNextChild

objParent.InsertChild objNewChild
Set objNewChild = Nothing
```

Los elementos secundario se deben insertar en determinado orden. Evite insertar atributos en la secuencia después de otros elementos secundarios. Es decir, los atributos no pueden tener elementos precedentes de ningún tipo y los demás elementos no pueden tener un elemento sucesorio de tipo atributo.

La vista Authentic exige un control especial de la presentación del valor de texto de un elemento XML, excepto en el caso de los atributos. El texto (o contenido) debe formar parte de un nodo secundario adicional de tipo texto. Este elemento se puede crear con [Authentic.CreateChild](#) y el valor de parámetro 6. En lugar de establecer el valor de texto del elemento directamente, debe establecer el valor de texto del nodo secundario.

Copiar objetos XMLData

Si quiere mover objetos XMLData a otra posición del mismo archivo, no puede usar los métodos [XMLData.InsertChild](#) y [XMLData.AppendChild](#). Estos métodos solo sirven para objetos XMLData nuevos.

En lugar de usar [InsertChild](#) o [AppendChild](#), debe copiar la jerarquía de objetos a mano. Por ejemplo, podría usar esta función escrita en JavaScript para copiar XMLData recursivamente:

```
// esta función devuelve una copia completa del objeto XMLData
function GetCopy(objXMLData)
{
```

```
var objNew;
objNew = objPlugIn.CreateChild(objXMLData.Kind);

objNew.Name = objXMLData.Name;
objNew.TextValue = objXMLData.TextValue;

if(objXMLData.HasChildren) {
    var objChild;
    objChild = objXMLData.GetFirstChild(-1);

    while(objChild) {
        try {
            objNew.AppendChild(GetCopy(objChild));
            objChild = objXMLData.GetNextChild();
        }
        catch(e) {
            objChild = null;
        }
    }

    return objNew;
}
```

Borrar objetos XMLData

XMLData ofrece dos métodos para la eliminación de objetos secundarios:

[XMLData.EraseAllChildren](#) y [XMLData.EraseCurrentChild](#).

Para borrar objetos XMLData debe tener acceso al primario de los elementos que desea eliminar.

Para obtener una referencia al objeto XMLData primario utilice [XMLData.GetFirstChild](#) y [XMLData.GetNextChild](#).

Para ver un ejemplo de cómo borrar elementos XML, consulte la descripción de los métodos

[XMLData.EraseAllChildren](#) y [XMLData.EraseCurrentChild](#).

1.14 DOM y XMLData

La interfaz XMLData ofrece total acceso a la estructura XML del documento actual con menos métodos que DOM y de manera mucho más sencilla. La interfaz XMLData ofrece una manera minimalista de leer y modificar datos XML nuevos o ya existentes. Sin embargo, puede usar la estructura DOM si quiere tener acceso desde una fuente de datos externa o si prefiere la implementación MSXML DOM.

Las funciones **ProcessDOMNode()** y **ProcessXMLDataNode()** que aparecen más abajo convierten los segmentos de una estructura XML de XMLData a DOM.

Para usar la función **ProcessDOMNode()**:

- pase el elemento raíz del segmento DOM que quiere convertir en **objNode** y
- pase el objeto del complemento con el método **CreateChild()** en **objCreator**

Para usar la función **ProcessXMLDataNode()**:

- pase el elemento raíz del segmento XMLData en **objXMLData** y
- pase el objeto DOMDocument creado con MSXML en **xmlDoc**

```
///////////////////////////////
// conversión de segmentos DOM en XMLData

function ProcessDOMNode(objNode,objCreator)
{
    var objRoot;
    objRoot = CreateXMLDataFromDOMNode(objNode,objCreator);

    if(objRoot) {
        if((objNode.nodeValue != null) && (objNode.nodeValue.length > 0))
            objRoot.TextValue = objNode.nodeValue;

        // agregar atributos
        if(objNode.attributes) {
            var Attribute;
            var oNodeList = objNode.attributes;

            for(var i = 0;i < oNodeList.length; i++) {
                Attribute = oNodeList.item(i);

                var newNode;
                newNode = ProcessDOMNode(Attribute,objCreator);

                objRoot.AppendChild(newNode);
            }
        }

        if(objNode.hasChildNodes) {
            try {
                // agregar secundarios
                var Item;
                oNodeList = objNode.childNodes;
            }
        }
    }
}
```

```
for(var i = 0;i < oNodeList.length; i++) {
    Item = oNodeList.item(i);

    var newNode;
    newNode = ProcessDOMNode(Item,objCreator);

    objRoot.appendChild(newNode);
}
}

catch(err) {
}

}

return objRoot;
}

function CreateXMLDataFromDOMNode(objNode,objCreator)
{
    var bSetName = true;
    var bSetValue = true;

    var nKind = 4;

    switch(objNode.nodeType) {
        case 2:nKind = 5;break;
        case 3:nKind = 6;bSetName = false;break;
        case 4:nKind = 7;bSetName = false;break;
        case 8:nKind = 8;bSetName = false;break;
        case 7:nKind = 9;break;
    }

    var objNew = null;
    objNew = objCreator.CreateChild(nKind);

    if(bSetName)
        objNew.Name = objNode.nodeName;

    if(bSetValue && (objNode.nodeValue != null))
        objNew.TextValue = objNode.nodeValue;

    return objNew;
}

///////////////////////////////
// conversión de segmentos XMLData en DOM

function ProcessXMLDataNode(objXMLData,xmlDoc)
{
    var objRoot;
    objRoot = CreateDOMNodeFromXMLData(objXMLData,xmlDoc);
```

```
if(objRoot) {
    if(IsTextNodeEnabled(objRoot) && (objXMLData.TextValue.length > 0))
        objRoot.appendChild(xmlDoc.createTextNode(objXMLData.TextValue));

    if(objXMLData.HasChildren) {
        try {
            var objChild;
            objChild = objXMLData.GetFirstChild(-1);

            while(true) {
                if(objChild) {
                    var newNode;
                    newNode = ProcessXMLDataNode(objChild, xmlDoc);

                    if(newNode.nodeType == 2) {
                        // el nodo secundario es un atributo
                        objRoot.attributes.setNamedItem(newNode);
                    }
                    else
                        objRoot.appendChild(newNode);
                }

                objChild = objXMLData.GetNextChild();
            }
        }
        catch(err) {
        }
    }
}

return objRoot;
}

function CreateDOMNodeFromXMLData(objXMLData, xmlDoc)
{
    switch(objXMLData.Kind) {
        case 4:return xmlDoc.createElement(objXMLData.Name);
        case 5:return xmlDoc.createAttribute(objXMLData.Name);
        case 6:return xmlDoc.createTextNode(objXMLData.TextValue);
        case 7:return xmlDoc.createCDATASection(objXMLData.TextValue);
        case 8:return xmlDoc.createComment(objXMLData.TextValue);
        case 9:return
            xmlDoc.createProcessingInstruction(objXMLData.Name, objXMLData.TextValue);
    }

    return xmlDoc.createElement(objXMLData.Name);
}

function IsTextNodeEnabled(objNode)
{
    switch(objNode.nodeType) {
```

```
case 1:  
case 2:  
case 5:  
case 6:  
case 11: return true;  
}  
  
return false;  
}
```

1.15 Scripting de Authentic

La función **Scripting de Authentic** permite crear diseños SPS más flexibles e interactivos. Estos diseños se pueden crear o editar en StyleVision Enterprise y StyleVision Professional y se pueden ver en la vista Authentic de los productos de Altova.

A continuación puede ver con qué productos de Altova es compatible esta función. No obstante, recuerde que en la versión segura del complemento Authentic Browser, los scripts internos se desactivan por motivos de seguridad.

Producto de Altova	¿Creación de scripts de Authentic?	¿Scripts de Authentic habilitados?
StyleVision Enterprise	Sí	Sí
StyleVision Professional	Sí	Sí
StyleVision Standard *	No	No
XMLSpy Enterprise	No	Sí
XMLSpy Professional	No	Sí
XMLSpy Standard	No	No
AuthenticDesktop Community	No	No
AuthenticDesktop Enterprise	No	Sí
Authentic Browser Plug-in Community **	No	No
Authentic Browser Plug-in Enterprise Trusted ***	No	Sí
Authentic Browser Plug-in Enterprise Untrusted	No	Sí

* No tiene la vista Authentic

** Tanto en la versión Trusted como en la versión Untrusted

*** Muestra los diseños que incluyen scripts. Incompatible con la ejecución de macros y con el control de eventos. Los eventos externos se desencadenan.

Los scripts de Authentic se comportan de la misma manera en todos los productos de Altova, por lo que no es necesaria una configuración ni un código especial para cada producto.

Cómo funcionan los scripts de Authentic

El autor del diseño SPS puede usar la función Scripting de Authentic de dos maneras:

- Puede asignar scripts a acciones definidas por el usuario (macros) para diseñar elementos, botones de barras de herramientas y comandos de menús contextuales.
- Puede añadir controladores de eventos al diseño que reaccionen a eventos de la vista Authentic.

Todos los scripts necesarios para convertir los documentos de Authentic en documentos interactivos se crean en StyleVision. Los formularios, las macros y los controladores de eventos se crean dentro del editor de scripts de StyleVision y los scripts se guardan con el archivo SPS. Después, en la vista Diseño de StyleVision, los scripts guardados se asignan a elementos de diseño, a botones de barras de herramientas o a menús contextuales. Cuando en la vista

Authentic de un producto de Altova compatible con scripts de Authentic se abre un documento XML basado en este diseño SPS, el documento tendrá las opciones interactivas creadas en el editor de scripts.

Documentación para scripts de Authentic

La documentación necesaria para aprender a trabajar con scripts de Authentic está incluida en la documentación de Altova StyleVision y se puede consultar en el [sitio web de Altova](#).

2 Objetos

Esta sección enumera y describe los objetos de Authentic Browser.

2.1 Authentic

Métodos

[StartEditing](#)

[LoadXML](#)

[Reset](#)

[Save](#)

[SavePOST](#)

[SaveXML](#)

[ValidateDocument](#)

[EditClear](#)

[EditCopy](#)

[EditCut](#)

[EditPaste](#)

[EditRedo](#)

[EditSelectAll](#)

[EditUndo](#)

[RowAppend](#)

[RowDelete](#)

[RowDuplicate](#)

[RowInsert](#)

[RowMoveDown](#)

[RowMoveUp](#)

[FindDialog](#)

[FindNext](#)

[ReplaceDialog](#)

[ApplyTextState](#)

[IsTextStateApplied](#)

[IsTextStateEnabled](#)

[MarkUpView](#)

[Print](#)

[PrintPreview](#)

[CreateChild](#)

[GetAllowedElements](#)

[GetAllAttributes](#)

[GetNextVisible](#)

[GetPreviousVisible](#)

[SelectionMoveTabOrder](#)

[SelectionSet](#)

[ClearSelection](#)

[attachCallBack](#)

[ReloadToolbars](#)

[StartSpellChecking](#)

[GetFileVersion](#)

[RedrawEntryHelpers](#)

[SetUnmodified](#)

[ClearUndoRedo](#)

Propiedades

[AuthenticView](#)

[IsEditClearEnabled](#)

[IsEditCopyEnabled](#)

[IsEditCutEnabled](#)

[IsEditPasteEnabled](#)

[IsEditRedoEnabled](#)

[IsEditUndoEnabled](#)

[IsFindNextEnabled](#)

[IsRowAppendEnabled](#)

[IsRowDeleteEnabled](#)

[IsRowDuplicateEnabled](#)

[IsRowInsertEnabled](#)

[IsRowMoveDownEnabled](#)

[IsRowMoveUpEnabled](#)

[SchemaLoadObject](#)

[XMLDataLoadObject](#)

[DesignDataLoadObject](#)

[XMLDataSaveUrl](#)

[XMLRoot](#)

[CurrentSelection](#)

[event](#)

[validationBadData](#)

[validationMessage](#)

[ToolbarsEnabled](#)

[ToolbarToolipsEnabled](#)

[AutoHideUnusedCommandGroups](#)

[TextStateToolbarLine](#)

[TextStateBmpURL](#)

[ToolbarRows](#)

[UICommands](#)

[XMLTable](#)

[BaseURL](#)

[EntryHelpersEnabled](#)

[EntryHelperSize](#)

[EntryHelperAlignment](#)

[EntryHelperWindows](#)

[EnableModifications](#)

[Modified](#)

[SaveButtonAutoEnable](#)

Eventos

[SelectionChanged](#)

[ControlInitialized](#)

Descripción

Clase de Authentic

2.1.1 Authentic.ApplyTextState

Declaración `ApplyTextState(elementName as String)`

Descripción

Aplica o quita el estado de texto definido por el parámetro `elementName`. Por ejemplo, el parámetro `elementName` podría ser `strong` (negrita) e `italic` (cursiva).

En un documento XML hay segmentos de datos que pueden contener subelementos. Por ejemplo:

```
<b>f r agmēnt </ b>
```

La etiqueta HTML `` hace que la palabra aparezca en negrita. Sin embargo, esto ocurre porque el analizador HTML sabe que la etiqueta `` corresponde al estilo negrita. Con XML puede tener mucha más flexibilidad: es posible definir cualquier etiqueta XML para lo que se quiera. Es decir, puede aplicar estado de texto usando XML, pero el estado de texto que se aplica debe formar parte del esquema.

Por ejemplo, en el ejemplo `OrgChart.xml`, `OrgChart.sps` y `OrgChart.xsd` la etiqueta `` es como la etiqueta para el estilo negrita. Para aplicar la negrita se llama al método `ApplyTextState()`. Pero al igual que las operaciones de filas y de edición, es necesario probar si se puede aplicar el estado de texto.

Consulte también [IsTextStateEnabled](#) y [IsTextStateApplied](#).

2.1.2 Authentic.attachCallBack

Obsoleto

En su lugar utilice los eventos de punto de conexión tal y como se describe [aquí](#).

Declaración attachCallBack(*bstrName* as String,*varCallBack* as Variant)

Descripción

La vista Authentic ofrece eventos que se pueden controlar por medio de funciones personales de devolución de llamada. Ningún controlador de eventos toma parámetros y se ignorará el valor devuelto. Para recuperar información cuando se genera un evento concreto debe leer las propiedades correspondientes del objeto del [evento](#).

Lista de eventos disponibles:

ondragover
ondrop
onkeydown
onkeyup
onkeypress
onmousemove
onbuttonup
onbuttondown
oneditpaste
oneditcut
oneditcopy

Desde la versión 3.0.0.0:

ondoceeditcommand

Desde la versión 5.3.0.0:

onbuttondoubleclick

Ejemplo de JavaScript:

```
// en alguna parte del script:  
objPlugIn.attachCallBack( "ondragover" ,OnDragOver);  
objPlugIn.attachCallBack( "ondrop" ,OnDrop);  
  
// controladores de eventos  
function OnDragOver()  
{  
    if(!objPlugIn.event.dataTransfer.ownDrag &&  
        objPlugIn.event.dataTransfer.type == "TEXT") )  
    {  
        objPlugIn.event.dataTransfer.dropEffect = 1;  
        objPlugIn.event.cancelBubble = true;  
    }  
}
```

```
// OnDrop() reemplaza el valor de texto completo del elemento XML
// con la selección de la operación de arrastrar
function OnDrop()
{
    var objTransfer = objPlugIn.event.dataTransfer;

    if(!objTransfer.ownDrag &&
       (objTransfer.type == "TEXT"))
        objPlugIn.event.srcElement.TextValue = objTransfer.getData();
}
```

2.1.3 AuthenticView

Declaración [AuthenticView](#) as [AuthenticView](#) (read-only)

Descripción

Devuelve un objeto que ofrece acceso a propiedades y métodos propios de la vista Authentic.

[AuthenticView](#) se solapa con las funciones concretas de la vista existente. Las versiones futuras de AuthenticView incluirá todas las funciones propias de la vista. El objeto AuthenticView es la interfaz recomendada para todas las futuras implementaciones.

Ejemplos

Consulte [Ordenar tablas](#) para obtener más información sobre cómo usar el objeto AuthenticView.

2.1.4 **Authentic.AutoHideUnusedCommandGroups**

Declaración AutoHideUnusedCommandGroups as Boolean

Descripción

True si es una barra de herramientas sin usar. Los grupos de comandos se ocultan automáticamente (p. ej. XML, los comandos de tabla, si el archivo SPS actual no admite el uso de tablas XML).

Valor predeterminado: true

2.1.5 **Authentic.BaseURL**

Declaración **BaseUrl** as **String**

Descripción

Esta propiedad establece la URL base para resolver rutas de acceso relativas. Si no se establece la URL, se utiliza la ubicación del archivo XML actual.

2.1.6 **Authentic.ClearSelection**

Declaración [ClearSelection\(\)](#)

Descripción

El método borra la selección actual. A partir de ese momento no se puede obtener la selección por medio de la propiedad CurrentSelection hasta que el usuario seleccione un nodo o se procese una llamada a [SelectionSet\(\)](#) correctamente.

Para poder borrar un nodo que tiene la selección actual, la selección debe establecerse en otro nodo o borrarse.

2.1.7 **Authentic.ClearUndoRedo**

Declaración [ClearUndoRedo\(\)](#)

Descripción

El método borra el búfer de las operaciones Deshacer/Rehacer.

2.1.8 **Authentic.ControlInitialized**

Declaración [ControlInitialized](#)

Descripción

Este evento se genera cuando se crea e inicializa el control.

Consulte también el apartado [Eventos de punto de conexión](#).

2.1.9 Authentic.CreateChild

Declaración CreateChild(*nKind* as [SPYXMLDataKind](#)) as [XMLData](#)

Valor devuelto

XML nodo nuevo

Descripción

El método CreateChild sirve para crear nodos nuevos que se pueden insertar en la estructura XML del documento actual usando la interfaz [XMLData](#).

Consulte también [XMLData.AppendChild](#) y [XMLData.InsertChild](#).

2.1.10 Authentic.CurrentSelection

Declaración CurrentSelection as [AuthenticSelection](#)

Descripción

La propiedad ofrece acceso a la selección actual en la vista Authentic.

El código del ejemplo siguiente recupera el texto completo de la selección actual:

JavaScript:

```
// en alguna parte del script:  
GetSelection(objPlugIn.CurrentSelection);  
  
// GetSelection() recopila la selección total del texto  
function GetSelection(objSel)  
{  
    var strText = "";  
  
    var objCurrent = objSel.Start;  
  
    while(!objSel.End.IsSameNode(objCurrent))  
    {  
        objCurrent = objPlugIn.GetNextVisible(objCurrent);  
        strText += objCurrent.TextValue;  
    }  
  
    strText += objSel.End.TextValue.substring(0,objSel.EndTextPosition);  
    return objSel.Start.TextValue.substr(objSel.StartTextPosition) + strText;  
}
```

2.1.11 Authentic.DesignDataLoadObject

See also

Declaración [DesignDataLoadObject](#) as [AuthenticLoadObject](#)

Descripción

DesignDataLoadObject contiene una referencia al documento SPS. El documento SPS sirve para generar el entorno de edición WYSIWYG y se genera por lo general con StyleVision.

Consulte [SchemaLoadObject](#) para ver un ejemplo.

2.1.12 Authentic.EditClear

Declaración [EditClear](#)

Descripción

Borra la selección actual.

2.1.13 Authentic.EditCopy

Declaración [EditCopy](#)

Descripción

Copia la selección actual en el portapapeles.

2.1.14 Authentic.EditCut

Declaración [EditCut](#)

Descripción

Corta la selección actual del documento y la copia en el portapapeles.

2.1.15 Authentic.EditPaste

Declaración [EditPaste](#)

Descripción

Pega el contenido del portapapeles en el documento.

2.1.16 Authentic.EditRedo

Declaración [EditRedo](#)

Descripción

Rehace el último paso deshacer.

2.1.17 Authentic.EditSelectAll

Declaración [EditSelectAll](#)

Descripción

El método selecciona el documento completo.

2.1.18 Authentic.EditUndo

Declaración [EditUndo](#)

Descripción

Deshace la última acción.

2.1.19 Authentic.EnableModifications

Declaración `EnableModifications` as Boolean

Descripción

True si está habilitada la modificación del contenido XML en Authentic Browser. Consulte también la propiedad [Modified](#).

Valor predeterminado: TRUE

2.1.20 Authentic.EntryHelperAlignment

Declaración EntryHelperAlignment as [SPYAuthenticToolbarAlignment](#)

Descripción

Esta propiedad se puede usar para establecer la posición de los ayudantes de entrada. El valor predeterminado es 3, que coloca los ayudantes de entrada en el lado derecho de la interfaz.

2.1.21 Authentic.EntryHelpersEnabled

Declaración EntryHelpersEnabled as Boolean

Descripción

True si están habilitados los ayudantes de entrada de Authentic Browser.

Esta propiedad se puede usar para habilitar o deshabilitar los ayudantes de entrada.

Valor predeterminado: FALSE

2.1.22 Authentic.EntryHelperSize

Declaración EntryHelperSize as Integer

Descripción

Esta propiedad se puede usar para establecer el tamaño inicial del área de los ayudantes de entrada en píxeles. Si quiere que se ignore el valor, asigne el valor -1 a la propiedad.

Valor predeterminado: -1

2.1.23 Authentic.EntryHelperWindows

Declaración EntryHelperWindows as [SPYAuthenticEntryHelperWindows](#)

Descripción

Esta propiedad se puede usar para definir qué ayudantes de entrada se muestran. Los valores se pueden combinar para mostrar más de un ayudante de entrada. El valor predeterminado para mostrar los tres ayudantes de entrada es 7.

2.1.24 Authentic.event

See also

Declaración event as [AuthenticEvent](#)

Descripción

Esta propiedad almacena un objeto de datos de evento que contiene información sobre el evento actual.

2.1.25 Authentic.FindDialog

See also

Declaración [FindDialog](#)

Descripción

Muestra el cuadro de diálogo "Buscar".

Consulte también el apartado [Buscar y reemplazar](#).

2.1.26 Authentic.FindNext

Declaración [FindNext](#)

Descripción

El método realiza la operación Buscar siguiente.

Consulte también el apartado [Buscar y reemplazar](#).

2.1.27 Authentic.GetAllAttributes

Declaración GetAllAttributes(*pForElement* as XMLData, *pElements* as Variant)

Descripción

GetAllAttributes() devuelve los atributos permitidos para el elemento indicado en forma de matriz de cadenas.

Ejemplo de JavaScript:

```
function GetAttributes()
{
    var arrElements = new Array(1);

    var objStart = objPlugIn.CurrentSelection.Start;

    var strText;
    strText = "Valid attributes at current selection:\n\n";

    for(var i = 1;i <= 4;i++)
    {
        objPlugIn.GetAllAttributes(objStart, arrElements);
        strText = strText + ListArray(arrElements) + "-----\n";
    }

    return strText;
}

function ListArray(arrIn)
{
    var strText = "";

    if(typeof(arrIn) == "object")
    {
        for(var i = 0;i <= (arrIn.length - 1);i++)
            strText = strText + arrIn[i] + "\n";
    }

    return strText;
}
```

Ejemplo de VBScript:

```
Sub DisplayAllowedAttributes
    dim arrElements()

    dim objStart
    dim objEnd
    set objStart = objPlugIn.CurrentSelection.Start
    set objEnd = objPlugIn.CurrentSelection.End

    dim strText
    strText = "Valid attributes at current selection:" & chr(13) & chr(13)
```

```
dim i

For i = 1 To 4
    objView.GetAllAttributes objStart, arrElements
    strText = strText & ListArray(arrElements) & "-----" & chr(13)
Next

msgbox strText
End Sub

Function ListArray(arrIn)
    dim strText

    If IsArray(arrIn) Then
        dim i

        For i = 0 To UBound(arrIn)
            strText = strText & arrIn(i) & chr(13)
        Next
    End If

    ListArray = strText
End Function
```

2.1.28 Authentic.GetAllowedElements

Declaración GetAllowedElements(*nAction* as SPYAuthenticElementActions, *pStartElement* as XMLData, *pEndElement* as XMLData, *pElements* as Variant)

Descripción

GetAllowedElements() devuelve los elementos permitidos para las acciones especificadas por nAction.

Ejemplo de JavaScript:

```
function GetAllowed()
{
    var arrElements = new Array(1);

    var objStart = objPlugIn.CurrentSelection.Start;
    var objEnd = objPlugIn.CurrentSelection.End;

    var strText;
    strText = "valid elements at current selection:\n\n";

    for(var i = 0;i <= 4;i++) {
        objPlugIn.GetAllowedElements(i,objStart,objEnd,arrElements);
        strText = strText + ListArray(arrElements) + "-----\n";
    }

    return strText;
}

function ListArray(arrIn)
{
    var strText = "";

    if(typeof(arrIn) == "object"){
        for(var i = 0;i <= (arrIn.length - 1);i++)
            strText = strText + arrIn[i] + "\n";
    }

    return strText;
}
```

Ejemplo de VBScript:

```
Sub DisplayAllowed
    dim arrElements()

    dim objStart
    dim objEnd
    set objStart = objPlugIn.CurrentSelection.Start
    set objEnd = objPlugIn.CurrentSelection.End
```

```
dim strText
strText = "Valid elements at current selection:" & chr(13) & chr(13)

dim i

For i = 1 To 4
    objView.GetAllowedElements i,objStart,objEnd,arrElements
    strText = strText & ListArray(arrElements) & "-----" & chr(13)
Next

msgbox strText
End Sub

Function ListArray(arrIn)
dim strText

If IsArray(arrIn) Then
    dim i

    For i = 0 To UBound(arrIn)
        strText = strText & arrIn(i) & chr(13)
    Next
End If

ListArray = strText
End Function
```

2.1.29 Authentic.GetFileVersion

Declaración GetFileVersion(*strVersion* as String)

Descripción

El método simplemente devuelve la versión del componente en forma de cadena, con el formato 5.0.0.0.

2.1.30 Authentic.GetNextVisible

Declaración GetNextVisible(*pElement* as XMLData) as XMLData

Descripción

El método obtiene el siguiente elemento XML visible del documento.

2.1.31 Authentic.GetPreviousVisible

Declaración GetPreviousVisible(*pElement* as [XMLData](#)) as [XMLData](#)

Descripción

El método obtiene el elemento XML visible anterior del documento.

2.1.32 Authentic.IsEditClearEnabled

Declaración IsEditClearEnabled as Boolean

Descripción

True si [EditClear](#) es posible.

Consulte también el apartado [Operaciones de edición](#).

2.1.33 Authentic.IsEditCopyEnabled

Declaración IsEditCopyEnabled as Boolean

Descripción

True si se puede copiar en el portapapeles.

Consulte también [EditCopy](#) y [Operaciones de edición](#).

2.1.34 Authentic.IsEditCutEnabled

Declaración IsEditCutEnabled as Boolean

Descripción

True si [EditCut](#) es posible en ese momento.

Consulte también el apartado [Operaciones de edición](#).

2.1.35 Authentic.IsEditPasteEnabled

Declaración IsEditPasteEnabled as Boolean

Descripción

True si [EditPaste](#) es posible.

Consulte también el apartado [Operaciones de edición](#).

2.1.36 Authentic.IsEditRedoEnabled

Declaración IsEditRedoEnabled as Boolean

Descripción

True si [EditRedo](#) es posible en ese momento.

Consulte también el apartado [Operaciones de edición](#).

2.1.37 Authentic.IsEditUndoEnabled

Declaración IsEditUndoEnabled as Boolean

Descripción

True si [EditUndo](#) es posible.

Consulte también el apartado [Operaciones de edición](#).

2.1.38 Authentic.IsFindNextEnabled

Declaración IsFindNextEnabled as Boolean

Descripción

True si FindNext es posible en ese momento. False si no quedan más instancias.

Consulte también el apartado [Buscar y reemplazar](#) y [FindDialog](#).

2.1.39 Authentic.IsRowAppendEnabled

Declaración IsRowAppendEnabled as Boolean

Descripción

True si [RowAppend](#) es posible.

Consulte también [Operaciones en filas](#).

2.1.40 Authentic.IsRowDeleteEnabled

Declaración IsRowDeleteEnabled as Boolean

Descripción

True si [RowDelete](#) es posible.

Consulte también [Operaciones en filas](#).

2.1.41 Authentic.IsRowDuplicateEnabled

Declaración IsRowDuplicateEnabled as Boolean

Descripción

True si [RowDuplicate](#) es posible en ese momento.

Consulte también [Operaciones en filas](#).

2.1.42 Authentic.IsRowInsertEnabled

Declaración IsRowInsertEnabled as Boolean

Descripción

True si [RowInsert](#) es posible.

Consulte también [Operaciones en filas](#).

2.1.43 Authentic.IsRowMoveDownEnabled

Declaración IsRowMoveDownEnabled as Boolean

Descripción

True si [RowMoveDown](#) es posible en ese momento.

Consulte también [Operaciones en filas](#).

2.1.44 Authentic.IsRowMoveUpEnabled

Declaración IsRowMoveUpEnabled as Boolean

Descripción

True si [RowMoveUp](#) es posible.

Consulte también [Operaciones en filas](#).

2.1.45 Authentic.IsTextStateApplied

Declaración IsTextStateApplied(*elementName* as String) as Boolean

Descripción

Comprueba si el estado del texto ya se ha aplicado. Ejemplos del parámetro *elementName* serían strong e italic.

2.1.46 Authentic.IsTextStateEnabled

Declaración IsTextStateEnabled(*elementName* as String) as Boolean

Descripción

Comprueba si se puede aplicar un estado del texto. Ejemplos del parámetro *elementName* serían strong e italic.

2.1.47 Authentic.LoadXML

Declaración LoadXML(*xmlString* as String)

Descripción

Carga el documento XML actual con la cadena XML aplicada. El contenido nuevo aparece inmediatamente.

2.1.48 Authentic.MarkUpView

Declaración `MarkUpView(kind as SPYAuthenticMarkupVisibility)`

Descripción

El documento se presenta por defecto usando técnicas HTML. Pero a veces es necesario ver las etiquetas de edición. Con este método puede mostrar las diferentes etiquetas de marcado.

2.1.49 Authentic.Modified

Declaración Modified as Boolean

Descripción

True si se modificó el contenido XML.

Esta propiedad es de solo lectura.

2.1.50 Authentic.Print

Declaración Print

Descripción

Imprimir el documento que se está editando.

2.1.51 Authentic.PrintPreview

Declaración [PrintPreview](#)

Descripción

Vista previa de impresión del documento que se está editando.

2.1.52 Authentic.RedrawEntryHelpers

Declaración [RedrawEntryHelpers\(\)](#)

Descripción

RedrawEntryHelpers toma los valores de las propiedades [EntryHelpersEnabled](#), [EntryHelperAlignment](#), [EntryHelperSize](#) y [EntryHelperWindows](#) y vuelve a dibujar las ventanas de los ayudantes de entrada.

2.1.53 Authentic.ReloadToolbars

Declaración ReloadToolbars()

Descripción

ReloadToolbars lee la colección [ToolbarRows](#) y vuelve a dibujar las barras de herramientas y la vista.

2.1.54 Authentic.ReplaceDialog

Declaración [ReplaceDialog](#)

Descripción

Muestra el cuadro de diálogo "Reemplazar".

Consulte también [Buscar y reemplazar](#).

2.1.55 Authentic.Reset

Obsoleto

Use [Authentic.StartEditing](#) en su lugar.

Declaración Reset

Descripción

Restaurar los datos que se están editando. Se le suele llamar antes de editar un conjunto nuevo de documentos XML, XSL y SPS.

El método no cambia la vista y todavía se puede seguir trabajando con el documento que está activo en la vista.

2.1.56 Authentic.RowAppend

Declaración [RowAppend](#)

Descripción

Anexa una fila en la posición actual.

Consulte también [Operaciones en filas](#).

2.1.57 Authentic.RowDelete

Declaración [RowDelete](#)

Descripción

Elimina las filas seleccionadas.

Consulte también [Operaciones en filas](#).

2.1.58 Authentic.RowDuplicate

Declaración [RowDuplicate](#)

Descripción

El método duplica las filas seleccionadas.

Consulte también [Operaciones en filas](#).

2.1.59 Authentic.RowInsert

Declaración RowInsert

Descripción

Inserta una fila nueva justo encima de la selección.

Consulte también [Operaciones en filas](#).

2.1.60 Authentic.RowMoveDown

Declaración [RowMoveDown](#)

Descripción

Baja la fila actual una posición.

Consulte también [Operaciones en filas](#).

2.1.61 Authentic.RowMoveUp

Declaración [RowMoveUp](#)

Descripción

Sube la fila actual una posición.

Consulte también [Operaciones en filas](#).

2.1.62 Authentic.Save

Declaración Save

Descripción

Guarda el documento en la URL indicada por la propiedad [XMLDataSaveUrl](#). En la versión no segura (*Untrusted*) también puede usar una ruta de acceso local completa.

El complemento envía una solicitud HTTP PUT al servidor para guardar el archivo XML activo en la vista.

2.1.63 Authentic.SaveButtonAutoEnable

Declaración SaveButtonAutoEnable as Boolean

Descripción

Si esta propiedad tiene el valor TRUE, el estado habilitado/deshabilitado del botón **Guardar** de la barra de herramientas del control se establece en función de la marca [Modified](#) del documento.

Valor predeterminado: FALSE.

2.1.64 Authentic.SavePOST

Declaración SavePOST

Descripción

Guarde el documento en la URL indicada por la propiedad [XMLDataSaveUrl](#). En la versión no segura (Untrusted) también puede usar una ruta de acceso local completa. El complemento envía una solicitud HTTP POST al servidor para guardar el archivo XML activo en la vista.

Comprobar si el archivo se guardó o no

Si el complemento Authentic Browser recibe una respuesta HTTP de ≥ 300 , entenderá que el archivo **no** se guardó y mostrará por defecto un mensaje con la respuesta de error HTTP y después otro mensaje informando al usuario de que el archivo no se guardó. El desarrollador de la aplicación debe asegurarse de que se devuelva la respuesta HTTP correcta. Por ejemplo puede usar este código PHP:

```
<?php
// suprimir mensajes de error para evitar
// que se generen resultados para poder enviar encabezados
error_reporting (0);
$error = false;
$handle = fopen ( "result.xml" , "w+" );
if (! $handle)
    $error = true;
else
{
    if (! fwrite($handle, $HTTP_RAW_POST_DATA))
        $error = true;
    else
        fclose($handle);
}
if ($error)
    header( "HTTP/1.1 500 Server Error" );
?>
```

2.1.65 Authentic.SaveXML

Declaración SaveXML as String

Valor devuelto

Estructura XML como cadena

Descripción

Guarda los datos XML actuales en una cadena de texto que se devuelve al autor de la llamada.

2.1.66 Authentic.SchemaLoadObject

Declaración SchemaLoadObject as [AuthenticLoadObject](#)

Descripción

SchemaLoadObject contiene una referencia al documento de esquema XML para el archivo XML actual. El documento de esquema se suele generar con Altova XMLSpy.

Ejemplo

```
objPlugIn.SchemaLoadObject.URL = "http://www.suservidor.com/OrgChart.xsd"
objPlugIn.XMLDataLoadObject.URL = "http://www.suservidor.com/OrgChart.xml"
objPlugIn.DesignDataLoadObject.URL = "http://www.suservidor.com/OrgChart.sps"
objPlugIn.StartEditing
```

El código del ejemplo anterior establece todas las propiedades URL de los objetos y llama [StartEditing](#) para cargar y mostrar los archivos. En la versión no segura (*Untrusted*) también puede usar una ruta de acceso local completa. El contenido actual y el estado del complemento se borrará.

2.1.67 Authentic.SelectionChanged

Declaración SelectionChanged as VT_0019

Descripción

Este evento se genera cada vez que el usuario cambia la selección actual.

Consulte también [Eventos del punto de conexión](#).

2.1.68 Authentic.SelectionMoveTabOrder

Declaración SelectionMoveTabOrder(*bForward* as Boolean,*bTag* as Boolean)

Descripción

SelectionMoveTabOrder() mueve hacia delante o hacia detrás la selección actual.

Si *bTag* es false y la selección actual está en la última celda de una tabla, entonces se añadirá una línea nueva.

2.1.69 Authentic.SelectionSet

*Declaración SelectionSet(*pStartElement* as [XMLData](#),*nStartPos* as long,*pEndElement* as [XMLData](#),*nEndPos* as long) as Boolean*

Descripción

Use SelectionSet() para establecer una selección nueva en la vista Authentic. Es posible dar a pEndElement el valor null (nada) si la selección debe estar justo encima de un elemento XML (StartElement).

2.1.70 Authentic.SetUnmodified

Declaración [SetUnmodified\(\)](#)

Descripción

Tras llamar a este método la condición actual del búfer de las operaciones Deshacer/Rehacer se toma como estado limpio del documento XML subyacente y la marca [Modified](#) toma el valor FALSE.

2.1.71 Authentic.StartEditing

Declaración StartEditing as Boolean

Valor devuelto

True si todos los archivos se cargaron y presentaron en la vista correctamente.

Descripción

Empezar a editar el documento actual. Es importante configurar primero las propiedades de los objetos [SchemaLoadObject](#), [DesignDataLoadObject](#) y [XMLDataLoadObject](#).

2.1.72 Authentic.StartSpellChecking

Declaración [StartSpellChecking\(\)](#)

Descripción

Este comando abre el cuadro de diálogo de corrección ortográfica si hay un paquete con un motor de corrección ortográfica y con los diccionarios necesarios y si ese paquete está activado.

2.1.73 Authentic.TextStateBmpURL

Declaración TextStateBmpURL as String

Descripción

La URL de la que se deben recuperar los mapas de bits para los iconos de estados del texto. Si no se indica ninguna URL, no habrá botones de estado del texto.

Ejemplos

```
objPlugIn.TextStateBmpURL = "<http://plugin.xmlspy.com/textstates/>"
```

```
<PARAM NAME="TextStateBmpURL" VALUE="http://plugin.xmlspy.com/textstates/">
```

2.1.74 Authentic.TextStateToolbarLine

Declaración TextStateToolbarLine as long

Descripción

La barra de herramientas (número de línea) en la que se deben colocar los iconos de los estados del texto. Estos iconos se pueden anexar a las barras de herramientas actuales o en barras de herramientas nuevas.

Valor predeterminado: 1

2.1.75 Authentic.ToolbarRows

Declaración ToolbarRows as [AuthenticToolbarRows](#)

Descripción

Obtiene una colección de todas las barras de herramientas que se deben mostrar. Consulte la descripción de [AuthenticToolbarRows](#), que explica cómo eliminar, añadir o modificar las barras de herramientas.

2.1.76 **Authentic.ToolbarsEnabled**

Declaración ToolbarsEnabled as Boolean

Descripción

True si las barras de herramientas de Authentic Browser están habilitadas.

Esta propiedad se puede usar para habilitar o deshabilitar todas las barras de herramientas.

Valor predeterminado: true

2.1.77 Authentic.ToolbarToolipsEnabled

Declaración ToolbarToolipsEnabled as Boolean

Descripción

True si está habilitada la información rápida de las barras de herramientas de Authentic Browser.

Valor predeterminado: true

2.1.78 Authentic.UICommands

Declaración UICommands as [AuthenticCommands](#)

Descripción

Obtiene una colección de todos los comandos de las barras de herramientas disponibles (y su descripción).

En las barras de herramientas se pueden colocar comandos para todos estos botones.

Solo lectura.

Ejemplo

Obtener todos los comandos, grupos de comandos y descripciones disponibles y presentarlos en un cuadro de mensaje.

```
dim str
for each UICommand in objPlugin.UICommands
str = str & UICommand.CommandID & " | " & UICommand.Group & " | " &
UICommand.ShortDescripción & chr(13)
next
msgbox str
```

2.1.79 Authentic.ValidateDocument

Declaración ValidateDocument(*showResults* as Boolean) as Boolean

Valor devuelto

Resultado de la validación

Descripción

Valida los datos XML actuales con los datos del esquema XML. Si el parámetro showResults es FALSE, entonces se suprimirán los errores de validación. De lo contrario se presentan los errores de validación.

2.1.80 Authentic.validationBadData

Declaración validationBadData as [XMLData](#)

Descripción

Esta propiedad puede aportar información adicional sobre el último error de validación. Se configura después de la llamada a ValidateDocument() y su valor puede ser null o una referencia al elemento XML que provocó el error.

2.1.81 Authentic.validationMessage

Declaración validationMessage as String

Descripción

Si falla la validación (después de la llamada a ValidateDocument), esta propiedad almacena una cadena de texto con el mensaje de error.

2.1.82 Authentic.XMLDataLoadObject

Declaración `XMLDataLoadObject` as [AuthenticLoadObject](#)

Descripción

`XMLDataLoadObject` contiene una referencia al documento XML que se está editando. El documento XML se suele definir con Altova XMLSpy, pero se suele generar con ayuda de una base de datos o cualquier otro proceso de negocio.

Consulte también [SchemaLoadObject](#) para ver un ejemplo.

2.1.83 Authentic.XMLDataSaveUrl

Declaración `XMLDataSaveUrl` as `String`

Descripción

Cuando los datos XML han sido modificados, se pueden volver a guardar en el servidor con una URL. Cuando se guardan a través de una solicitud `HTTP PUT` con ayuda del método `Authentic.Save`, esta propiedad define la ubicación donde se guardarán los datos XML. Cuando se envían a través de una solicitud `HTTP POST` con ayuda del método `Authentic.SavePOST`, esta propiedad defina la ubicación de una aplicación o de un script del lado servidor que procesará los datos de `POST`. En la versión no segura (*Untrusted*) también puede usar una ruta de acceso local completa.

Consulte también los métodos [Authentic.Save](#) y [Authentic.SavePOST](#).

2.1.84 Authentic.XMLRoot

Declaración XMLRoot as XMLData

Descripción

XMLRoot es el elemento primario de la estructura XML activa en la vista. La interfaz [XMLData](#) otorga acceso a todo el contenido del archivo.

Consulte también [Using XMLData](#).

2.1.85 Authentic.XMLTable

Declaración XMLTable as [AuthenticXMLTableCommands](#)

Descripción

Obtiene el conjunto formado por todos los comandos de tabla XML.
Solo lectura.

2.2 AuthenticCommand

Métodos

Propiedades

[CommandID](#)

[Group](#)

[ShortDescripción](#)

[Name](#)

2.2.1 AuthenticCommand.CommandID

Declaración CommandID as [SPYAuthenticCommand](#)

Descripción

Obtiene el CommandId del comando.

Valores posibles: véase AuthenticToolbarButton

Solo lectura

Ejemplo

Consulte el ejemplo de [Authentic.UICommands](#)

2.2.2 AuthenticCommand.Group

Declaración Group as [SPYAuthenticCommandGroup](#)

Descripción

El grupo de comandos al que pertenece el comando.

Solo lectura

Ejemplo

Consulte el ejemplo de [Authentic.UICommands](#)

2.2.3 **AuthenticCommand.ShortDescription**

Declaración ShortDescripción as String

Descripción

Breve descripción del comando (p. ej. el texto de información rápida)

Solo lectura

Ejemplo

Consulte el ejemplo de [Authentic.UICommands](#)

2.2.4 **AuthenticCommand.Name**

Declaración Name as String

Descripción

Para su uso posterior

2.3 AuthenticCommands

Métodos

[Item](#)

Propiedades

[Count](#)

2.3.1 **AuthenticCommands.Count**

Declaración Count as long

Descripción

Número de comandos disponibles en la interfaz del usuario

Solo lectura

2.3.2 AuthenticCommands.Item

Declaración Item (nPosition as long) as [AuthenticCommand](#)

Descripción

Obtiene el comando de la posición nPosition. nPosition empieza por el número 1.

2.4 AuthenticContextMenu

La interfaz `ContextMenu` permite al usuario personalizar los menús contextuales que aparecen en `Authentic`. La interfaz tiene los métodos que se enumeran en esta sección.

2.4.1 CountItems

Método: [CountItems\(\)](#)

Valor devuelto

Devuelve el número de elementos del menú.

Errores

2501 Objeto no válido.

2.4.2 DeleteItem

Método: `DeleteItem(position as integer)`

Valor devuelto

Elimina un elemento de menú.

Errores

- 2501 Objeto no válido
- 2502 Índice no válido

2.4.3 GetItemText

Método: `GetItemText(position as integer)` nombre del elemento de menú como cadena de texto

Valor devuelto

Obtiene el nombre del elemento del menú.

Errores

- 2501 Objeto no válido
- 2502 Índice no válido

2.4.4 InsertItem

Método: `InsertItem(position as integer, menu item name as string, macro name as string)`

Valor devuelto

Inserta un elemento de menú definido por el usuario. El elemento de menú iniciará una macro, por lo que es necesario suministrar un nombre de macro válido.

Errores

- 2501 Objeto no válido
- 2502 Índice no válido
- 2503 La macro no existe
- 2504 Error interno

2.4.5 SetItemText

Método: `SetItemText`(position as integer, menu item name as string)

Valor devuelto

Establece el nombre del elemento de menú.

Errores

- 2501 Objeto no válido
- 2502 Índice no válido

2.5 AuthenticDataTransfer

Métodos

[getData](#)

Propiedades

[dropEffect](#)

[ownDrag](#)

[type](#)

Descripción

Interfaz AuthenticDataTransfer.

2.5.1 AuthenticDataTransfer.dropEffect

Declaración dropEffect as long

Descripción

La propiedad almacena el efecto de la operación colocar del controlador de evento predeterminado. Este efecto de la operación de colocar se puede configurar cambiando este valor y asignando el valor TRUE a [AuthenticEvent.cancelBubble](#).

2.5.2 AuthenticDataTransfer.getData

Declaración `getData` as Variant

Descripción

`getData` obtiene los datos reales asociados con este objeto `dataTransfer`. Consulte también [AuthenticDataTransfer.type](#).

2.5.3 **AuthenticDataTransfer.ownDrag**

Declaración ownDrag as Boolean

Descripción

La propiedad tiene el valor TRUE si el origen de la operación de arrastrar actual viene de la vista Authentic.

2.5.4 **AuthenticDataTransfer.type**

Declaración type as String

Descripción

Almacena el tipo de datos que se obtienen con el método [AuthenticDataTransfer.getData](#).

Los tipos de datos compatibles actualmente son:

OWN	datos del propio complemento
TEXT	texto sin formato
UNICODETEXT	texto sin formato como UNICODE
IUNKNOWN	referencia IUnknown a la instancia IDataObject

2.6 AuthenticEvent

Propiedades

[altKey](#)

[altLeft](#)

[ctrlKey](#)

[ctrlLeft](#)

[shiftKey](#)

[shiftLeft](#)

[keyCode](#)

[repeat](#)

[button](#)

[clientX](#)

[clientY](#)

[dataTransfer](#)

[srcElement](#)

[fromElement](#)

[propertyName](#)

[cancelBubble](#)

[returnValue](#)

[type](#)

Descripción

Interfaz AuthenticEvent.

2.6.1 AuthenticEvent.altKey

Declaración altKey as Boolean

Descripción

True si se pula la tecla **Alt** derecha.

2.6.2 AuthenticEvent.altLeft

Declaración altLeft as Boolean

Descripción

True si se pulsa la tecla **Alt** izquierda.

2.6.3 AuthenticEvent.button

Declaración button as long

Descripción

Especifica qué botón del mouse se pulsó:

- 0 Ninguno
- 1 El botón izquierdo
- 2 El botón derecho
- 3 Se pulsó el botón izquierdo y el botón derecho a la vez
- 4 El botón del medio
- 5 Se pulsó el botón izquierdo y el botón del medio a la vez
- 6 Se pulsó el botón derecho y el botón del medio a la vez
- 7 Todos los botones a la vez

Los eventos onbuttondown y onbuttonup configuran el valor del botón. El evento onbuttonup configura el valor del botón que se soltó y genera el evento up independientemente de qué otros botones estén pulsados en ese momento.

2.6.4 AuthenticEvent.cancelBubble

Declaración cancelBubble as Boolean

Descripción

Define el valor TRUE para cancelBubble si no se debe llamar al controlador de eventos predeterminado.

2.6.5 AuthenticEvent.clientX

Declaración clientX as long

Descripción

Valor X de la posición actual del mouse en las coordenadas del cliente.

2.6.6 AuthenticEvent.clientY

Declaración clientY as long

Descripción

El valor Y de la posición actual del mouse en las coordenadas del cliente.

2.6.7 AuthenticEvent.ctrlKey

Declaración ctrlKey as Boolean

Descripción

True si se pulsa la tecla **Ctrl** derecha.

2.6.8 AuthenticEvent.ctrlLeft

Declaración ctrlLeft as Boolean

Descripción

True si se pulsa la tecla **Ctrl** izquierda.

2.6.9 AuthenticEvent.dataTransfer

Declaración `dataTransfer` as `Variant`

Descripción

propiedad `dataTransfer`

2.6.10 AuthenticEvent.fromElement

Declaración fromElement as Variant

Descripción

Ningún evento establece esta propiedad

2.6.11 AuthenticEvent.keyCode

Declaración keyCode as long

Descripción

Código clave de la tecla que está pulsada.

La propiedad es de lectura-escritura.

2.6.12 AuthenticEvent.propertyName

Declaración propertyName as String

Descripción

Ahora mismo no hay ningún evento que establezca esta propiedad.

2.6.13 AuthenticEvent.repeat

Declaración repeat as Boolean

Descripción

True si se repite el evento onkeydown.

2.6.14 AuthenticEvent.returnValue

Declaración returnValue as Variant

Descripción

Use returnValue para definir un valor devuelto para su controlador de eventos.

2.6.15 AuthenticEvent.shiftKey

Declaración shiftKey as Boolean

Descripción

True si se pulsa la tecla **Mayús** derecha.

2.6.16 AuthenticEvent.shiftLeft

Declaración shiftLeft as Boolean

Descripción

True si se pulsa la tecla **Mayús** izquierda.

2.6.17 AuthenticEvent.srcElement

Declaración `srcElement` as `Variant`

Descripción

Elemento que desencadena el evento actual.

Suele ser un objeto [XMLData](#).

Desde la versión 3.0.0.0: la propiedad también puede almacenar una referencia a un objeto [AuthenticCommand](#) si se estableció desde un evento ondoceditcommand.

2.6.18 AuthenticEvent.type

Declaración type as String

Descripción

Ahora mismo no hay ningún evento que establezca esta propiedad.

2.7 AuthenticEventContext

La interfaz `EventContext` ofrece acceso a muchas propiedades del contexto donde se ejecuta una macro.

2.7.1 EvaluateXPath

Método: `EvaluateXPath (string expression) as string`

Valor devuelto

El método evalúa la expresión XPath en el contexto del nodo en el que se desencadenó el evento y devuelve una cadena de texto.

Descripción

`EvaluateXPath()` ejecuta una expresión Xpath con el contexto del evento dado. El resultado se devuelve como cadena de texto y, si se trata de una secuencia, entonces es una cadena de texto separada por espacios.

Errores

- 2201 Objeto no válido.
- 2202 No hay contexto.
- 2209 Parámetro no válido.
- 2210 Error interno.
- 2211 Error de XPath.

2.7.2 GetEventType

Método: `GetEventType() as EventType enumeration`

Valor devuelto

Devuelve el tipo de nodo de contexto.

Descripción

`GetEventType` permite al usuario determinar si la macro está en un nodo XML o en un contexto de un elemento atómico XPath. La enumeración `AuthenticEventType` se define de la siguiente manera:

```
authenticEventContextXML,  
authenticEventContextAtomicItem,  
authenticEventContextOther
```

Si el contexto es un nodo XML normal, la función `GetXmlNode()` da acceso (de lo contrario devuelve `NULL`).

Errores

- 2201 Objeto no válido.
- 2202 No hay contexto.
- 2209 Parámetro no válido.

2.7.3 GetNormalizedTextValue

Método: `GetNormalizedTextValue()` value as string

Valor devuelto

Devuelve el valor del nodo actual en forma de cadena.

Errores

- 2201 Objeto no válido.
- 2202 No hay contexto.
- 2203 Contexto no válido
- 2209 Parámetro no válido.

2.7.4 GetVariableValue

Método: `GetVariableValue(nombre como cadena, valor como cadena)`

Valor devuelto

Obtiene el valor de la variable nombrada.

Descripción

`GetVariableValue` obtiene el valor de la variable en el ámbito del contexto.

```
nZoom = parseInt( AuthenticView.EventContext.GetVariableValue( 'Zoom' ) );
if ( nZoom > 1 )
{
    AuthenticView.EventContext.SetVariableValue( 'Zoom', nZoom - 1 );
}
```

Errores

- 2201 Objeto no válido.
- 2202 No hay contexto.
- 2204 Variable fuera de intervalo
- 2205 La variable no se puede evaluar
- 2206 La variable devuelve una secuencia
- 2209 Parámetro no válido

2.7.5 GetXMLNode

Método: [GetXMLNode\(\)](#) [XMLData](#) object

Valor devuelto

Devuelve el nodo XML de contexto o NULL

Errores

- 2201 Objeto no válido.
- 2202 No hay contexto.
- 2203 Contexto no válido
- 2209 Parámetro no válido.

2.7.6 IsAvailable

Método: `IsAvailable()`

Valor devuelto

Devuelve true si se estableció `EventContext`.

Errores

2201 Objeto no válido.

2.7.7 SetVariableValue

Método: `SetVariableValue(nombre como cadena, valor como cadena)`

Valor devuelto

Establece el valor de la variable nombrada.

Descripción

`SetVariableValue` establece el valor de la variable en el ámbito del contexto.

```
nZoom = parseInt( AuthenticView.EventContext.GetVariableValue( 'Zoom' ) );
if ( nZoom > 1 )
{
    AuthenticView.EventContext.SetVariableValue( 'Zoom', nZoom - 1 );
}
```

Errores

- 2201 Objeto no válido.
- 2202 No hay contexto.
- 2204 Variable fuera de intervalo
- 2205 La variable no se puede evaluar
- 2206 La variable devuelve una secuencia
- 2207 Variable de solo lectura
- 2208 No se permiten cambios

2.8 AuthenticLoadObject

Propiedades

[String](#)

[URL](#)

Descripción

El objeto XMLSpyXMLLoadSave sirve para establecer el origen de los archivos que se deben cargar. Puede establecer el contenido a través de la propiedad String directamente o como ubicación externa a través de la propiedad URL.

Para más información consulte [Authentic.SchemaLoadObject](#), [Authentic.DesignDataLoadObject](#) y [Authentic.XMLDataLoadObject](#).

2.8.1 **AuthenticLoadObject.String**

Declaración `String` as `String`

Descripción

Puede usar esta propiedad para establecer la estructura XML a partir de una cadena. La propiedad URL del objeto debe estar vacía si quiere usar esta propiedad.

2.8.2 AuthenticLoadObject.URL

Declaración URL as String

Descripción

La propiedad debería contener una URL válida para la operación Cargar o Guardar. Los protocolos HTTP compatibles son http, https, ftp y gopher.

2.9 AuthenticRange

La primera tabla que aparece a continuación enumera las propiedades y los métodos de AuthenticRange que se pueden usar para navegar por el documento y seleccionar partes concretas.

Propiedades	Métodos
Application	Clone
FirstTextPosition	MoveBegin
FirstXMLData	MoveEnd
FirstXMLDataOffset	NextCursorPosition
LastTextPosition	PreviousCursorPosition
LastXMLData	Select
LastXMLDataOffset	SelectNext
Parent	SelectPrevious
	SetFromRange
	IsEmpty
	IsEqual

Esta tabla enumera los métodos de modificación de contenido. La mayoría puede encontrarse en el menú que aparece al hacer clic con el botón secundario del mouse.

Propiedades	Operaciones de edición	Operaciones de tablas dinámicas
Text	Copy Cut Delete Paste	AppendRow DeleteRow DuplicateRow InsertRow IsInDynamicTable MoveRowDown MoveRowUp

Estos métodos ofrecen las funciones de los ayudantes de entrada de Authentic para los objetos de intervalo.

Operaciones de los ayudantes de entrada		
Elementos	Atributos	Entidades
CanPerformActionWith	GetElementAttributeValue	GetEntityNames
CanPerformAction	GetElementAttributeNames	InsertEntity
PerformAction	GetElementHierarchy	
	HasElementAttribute	
	SetElementAttributeValue	

Descripción

Los objetos de AuthenticRange son las selecciones realizadas con el cursor de la interfaz de automatización. Puede usarlos para apuntar a cualquier selección realizada con el cursor en la vista Authentic o para seleccionar parte del documento. Después las operaciones disponibles para los objetos de AuthenticRange trabajan en la selección de la misma manera que las operaciones correspondientes de la interfaz del usuario en la selección actual de la interfaz del usuario. La principal diferencia es que puede usar un número aleatorio de objetos de AuthenticRange al mismo tiempo, mientras que en la interfaz del usuario hay una sola selección realizada con el cursor.

Para obtener un objeto de intervalo inicial utilice [AuthenticView.Selection](#), con lo que obtendrá el intervalo que se corresponde con la selección realizada con el cursor en la interfaz del usuario. Otra opción es acceder a intervalos de segundo orden con las propiedades de solo lectura [AuthenticView.DocumentBegin](#), [AuthenticView.DocumentEnd](#) y [AuthenticView.WholeDocument](#). El método más flexible es [AuthenticView.Goto](#), que permite navegar hasta una parte concreta del documento dentro de una llamada. Para realizar selecciones más complejas, puede combinar las propiedades con los métodos de navegación en los objetos de intervalo que aparecen en la primera tabla de este apartado.

Otro método para seleccionar una porción del documento consiste en usar las propiedades de posición del objeto de intervalo. Existen dos sistemas de posicionamiento y se pueden combinar de forma aleatoria:

- **Posiciones absolutas del cursor de texto:** empezando por la posición 0 al principio del documento. Se puede establecer y recuperar para el principio y el final del intervalo. Para más información consulte [FirstTextPosition](#) y [LastTextPosition](#). Este método requiere cálculos internos complejos y debería utilizarse con cuidado.
- El elemento **XMLData** y una posición de texto dentro de este elemento: se puede establecer y recuperar para el principio y el final del intervalo. Para más información consulte [FirstXMLData](#), [FirstXMLDataOffset](#), [LastXMLData](#) y [LastXMLDataOffset](#). Este método es muy eficaz pero requiere conocimientos previos sobre la estructura subyacente del documento. Se puede utilizar para ubicar objetos XMLData y realizar operaciones en ellos que no están disponibles desde la interfaz del usuario.

Se pueden realizar modificaciones en el contenido del documento de varias maneras:

- La propiedad [Text](#) permite recuperar el texto del documento seleccionado por el objeto de intervalo. Si se le asigna un valor, el texto seleccionado del documento se reemplaza con el texto nuevo.
- Las funciones estándar de edición [Cortar](#), [Copiar](#), [Pegar](#) y [Eliminar](#).
- Operaciones de tabla para tabla que pueden crecer de forma dinámica.
- Métodos que crean asignaciones entre las funciones de los ayudantes de entrada de Authentic.
- Acceso a los objetos [XMLData](#) del documento subyacente para modificarlos directamente.

2.9.1 AuthenticRange.AppendRow

Método: [AppendRow \(\) as Boolean](#)

Descripción

Si el principio del intervalo está dentro de una tabla dinámica, este método inserta una fila nueva al final de la tabla seleccionada. La selección del intervalo se modifica para apuntar al principio de la nueva fila. La función devuelve *true* si la operación de anexión finaliza correctamente. De lo contrario devuelve *false*.

Errores

- 2001 El objeto de intervalo o su objeto relacionado ya no es válido.
- 2005 La dirección indicada para el parámetro de devolución no es válida.

Ejemplos

```
' -----  
'           VBScript  
' Anexar fila al final de la tabla dinámica actual  
' -----  
  
Dim objRange  
Set objRange = objPlugin.ActiveDocument.AuthenticView.Selection  
  
' comprobar si se puede insertar algo  
If objRange.IsInDynamicTable Then  
    objRange.AppendRow  
    ' objRange apunta al principio de la nueva fila  
    objRange.Select  
End If
```

2.9.2 AuthenticRange.Application

Propiedad: Application as [Authentic](#) (read-only)

Descripción

Acceder al objeto de aplicación de XMLSpy.

Errores

- 2001 El objeto de intervalo o su objeto relacionado ya no es válido.
- 2005 La dirección indicada para el parámetro de devolución no es válida.

2.9.3 AuthenticRange.CanPerformAction

Método: `CanPerformAction (eAction as SPYAuthenticActions, strElementName as String) as Boolean`

Descripción

`CanPerformAction` y sus métodos permiten el acceso a las funciones de los ayudantes de entrada de `Authentic`. Esta función permite modificar el contenido del documento de forma sencilla y coherente sin necesidad de saber dónde tienen lugar los cambios exactamente. El principio del objeto de intervalo se usa para encontrar la siguiente ubicación válida donde se puede llevar a cabo la acción indicada. Si se encuentra la ubicación, el método devuelve `True`. De lo contrario devuelve `False`.

CONSEJO: utilice [CanPerformActionWith](#) para encontrar todos los nombres de elemento válidos para una acción determinada.

Errores

- 2001 El objeto de intervalo o su objeto relacionado ya no es válido.
- 2005 La dirección indicada para el parámetro de devolución no es válida.
- 2007 La acción indicada no es válida.

Ejemplos

Consulte también [PerformAction](#).

2.9.4 AuthenticRange.CanPerformActionWith

Método: `CanPerformActionWith (eAction as SPYAuthenticActions, out_arrElementNames as Variant)`

Descripción

`CanPerformActionWith` y sus métodos permiten acceder a las funciones de los ayudantes de entrada de Authentic. Esta función permite modificar el contenido del documento de forma sencilla y coherente sin necesidad de saber dónde tienen lugar los cambios exactamente.

Este método devuelve una serie de nombres de elementos con los que se puede llevar a cabo la acción especificada.

CONSEJO: utilice [PerformAction](#) para aplicar la acción.

Errores

- 2001 El objeto de intervalo o su objeto relacionado ya no es válido.
- 2005 La dirección indicada para el parámetro de devolución no es válida.
- 2007 La acción indicada no es válida.

Ejemplos

Consulte también [PerformAction](#).

2.9.5 AuthenticRange.Clone

Método: [Clone \(\) as AuthenticRange](#)

Descripción

Devuelve una copia del objeto de intervalo.

Errores

- 2001 El objeto de intervalo o su objeto relacionado ya no es válido.
- 2005 La dirección indicada para el parámetro de devolución no es válida.

2.9.6 AuthenticRange.CollapsToBegin

Método: [CollapsToBegin \(\) as AuthenticRange](#)

Descripción

Establece el final del objeto de intervalo en su inicio. El método devuelve el objeto de intervalo modificado.

Errores

- 2001 El objeto de intervalo o su objeto relacionado ya no es válido.
- 2005 La dirección indicada para el parámetro de devolución no es válida.

2.9.7 AuthenticRange.CollapsToEnd

Método: [CollapsToEnd \(\) as AuthenticRange](#)

Descripción

Establece el principio del objeto de intervalo en su final. El método devuelve el objeto de intervalo modificado.

Errores

- 2001 El objeto de intervalo o su objeto relacionado ya no es válido.
- 2005 La dirección indicada para el parámetro de devolución no es válida.

2.9.8 AuthenticRange.Copy

Método: `Copy () as Boolean`

Descripción

Devuelve *False* si el intervalo no contiene porciones del documento que no se puedan copiar.

Devuelve *True* si se copió texto (y si hay elementos XML seleccionados, elementos también) en el búfer de copia/pega.

Errores

- 2001 El objeto de intervalo o su objeto relacionado ya no es válido.
- 2005 La dirección indicada para el parámetro de devolución no es válida.

2.9.9 AuthenticRange.Cut

Método: [Cut \(\)](#) as Boolean

Descripción

Devuelve *False* si el intervalo no contiene porciones del documento que no se pueden eliminar.

Devuelve *True* después de que se elimine texto (y si hay elementos XML seleccionados, elementos también) y se guarde en el búfer de copia/pega.

Errores

- 2001 El objeto de intervalo o su objeto relacionado ya no es válido.
- 2005 La dirección indicada para el parámetro de devolución no es válida.

2.9.10 AuthenticRange.Delete

Método: `Delete () as Boolean`

Descripción

Devuelve `False` si el intervalo no contiene porciones del documento que no se pueden eliminar.

Devuelve `True` después de que se elimine texto (y si hay elementos XML seleccionados, elementos también) del documento.

Errores

- 2001 El objeto de intervalo o su objeto relacionado ya no es válido.
- 2005 La dirección indicada para el parámetro de devolución no es válida.

2.9.11 AuthenticRange.DeleteRow

Método: `DeleteRow () as Boolean`

Descripción

Si el principio del intervalo está dentro de una tabla dinámica, este método elimina la fila seleccionada. La selección del intervalo se modifica para apuntar al siguiente elemento situado después de la fila eliminada. La función devuelve *true* si la operación de eliminar se lleva a cabo correctamente. De lo contrario devuelve *false*.

Errores

- 2001 El objeto de intervalo o su objeto relacionado ya no es válido.
- 2005 La dirección indicada para el parámetro de devolución no es válida.

Ejemplos

```
' -----  
'           VBScript  
' Eliminar fila seleccionada de la tabla dinámica  
' -----  
  
Dim objRange  
Set objRange = objPlugin.ActiveDocument.AuthenticView.Selection  
  
' comprobar si estamos en una tabla  
If objRange.IsInDynamicTable Then  
    objRange.DeleteRow  
End If
```

2.9.12 AuthenticRange.DuplicateRow

Método: `DuplicateRow () as Boolean`

Descripción

Si el principio del intervalo está dentro de una tabla dinámica, este método inserta una copia de la fila actual después de la fila seleccionada. La selección del intervalo se modifica para apuntar al principio de la nueva fila. La función devuelve *true* si la operación se completa correctamente. De lo contrario devuelve *false*.

Errores

- 2001 El objeto de intervalo o su objeto relacionado ya no es válido.
- 2005 La dirección indicada para el parámetro de devolución no es válida.

Ejemplos

```
' -----  
'           VBScript  
'   crear copia de la fila en la tabla dinámica actual  
' -----  
  
Dim objRange  
Set objRange = objPlugin.ActiveDocument.AuthenticView.Selection  
  
' comprobar si se puede insertar algo  
If objRange.IsInDynamicTable Then  
    objRange.DuplicateRow  
    ' objRange apunta al principio de la nueva fila  
    objRange.Select  
End If
```

2.9.13 AuthenticRange.EvaluateXPath

Método: [EvaluateXPath](#) (expresión de cadena) como cadena de texto

Valor devuelto

Este método devuelve una cadena de texto

Descripción

`EvaluateXPath()` ejecuta una expresión XPath con el nodo de contexto que está al principio de la selección del intervalo. El resultado se devuelve como cadena de texto y, si se trata de una secuencia, como cadena separada por espacios. Si el nodo de contexto XML es irrelevante, el usuario puede indicar cualquier otro nodo, como `AuthenticView.XMLDataRoot` por ejemplo.

Errores

- 2001 Objeto no válido
- 2005 Parámetro no válido
- 2008 Error interno
- 2202 Falta el nodo de contexto
- 2211 Error de XPath

2.9.14 AuthenticRange.ExpandTo

Método: `ExpandTo (eKind as SPYAuthenticElementKind) as AuthenticRange`

Descripción

Selecciona todo el elemento de tipo *eKind*, que empieza en o contiene la primera posición del cursor del intervalo. El método devuelve el objeto de intervalo modificado.

Errores

- 2001 El objeto de intervalo o su objeto relacionado ya no es válido.
- 2003 La expansión del intervalo sobrepasaría el final del documento.
- 2005 La dirección indicada para el parámetro de devolución no es válida.

2.9.15 AuthenticRange.FirstTextPosition

Propiedad: [FirstTextPosition](#) as Long

Descripción

Establecer u obtener el índice de posición del texto situado más a la izquierda del objeto de intervalo. Este índice siempre es menor o igual que [LastTextPosition](#). La indización empieza por 0 al principio del documento e incrementa con cada nueva posición que pueda ocupar el cursor de texto. Incrementar la posición de prueba 1 punto tiene el mismo efecto que la tecla a la derecha del cursor. Reducir la posición de prueba 1 punto tiene el mismo efecto que la tecla a la izquierda del cursor.

Si para *FirstTextPosition* define un valor mayor que el actual [LastTextPosition](#), [LastTextPosition](#) se establece en el nuevo *FirstTextPosition*.

CONSEJO: tenga cuidado al utilizar las posiciones del cursor de texto porque es una operación costosa comparada con el posicionamiento del cursor basado en XMLData.

Errores

- 2001 El objeto de intervalo o su objeto relacionado no es válido.
- 2005 La dirección indicada para el parámetro de devolución no es válida.
- 2006 Se indicó una posición del texto situada fuera del documento.

Ejemplos

```
' -----  
'           VBScript  
' -----  
  
Dim objAuthenticView  
' imaginamos que el documento activo está abierto en la vista Authentic  
Set objAuthenticView = objPlugin.AuthenticView  
  
nDocStartPosition = objAuthenticView.DocumentBegin.FirstTextPosition  
nDocEndPosition = objAuthenticView.DocumentEnd.FirstTextPosition  
  
' creamos un intervalo que selecciona todo el documento  
' de manera eficaz  
Dim objRange  
' primero tenemos que obtener un objeto de intervalo (cualquiera)  
Set objRange = objAuthenticView.DocumentBegin  
objRange.FirstTextPosition = nDocStartPosition  
objRange.LastTextPosition = nDocEndPosition  
  
' comprobamos si lo hemos hecho correctamente  
If objRange.isEqual(objAuthenticView.WholeDocument) Then  
    MsgBox "Funcionó la prueba usando posicionamiento directo del cursor de texto"  
Else  
    MsgBox "Ooops!"  
End If
```

2.9.16 AuthenticRange.FirstXMLData

Propiedad: FirstXMLData as XMLData

Descripción

Establecer u obtener el primer elemento XMLData del documento subyacente que esté parcial o completamente seleccionado por el intervalo. El principio exacto de la selección viene definido por el atributo [FirstXMLDataOffset](#).

Cuando establezca *FirstXMLData* en un objeto de datos nuevo, [FirstXMLDataOffset](#) se establece en la primera posición del cursor dentro de este elemento. Solamente se pueden utilizar objetos XMLData que tengan una posición de cursor. Si establece *FirstXMLData* / [FirstXMLDataOffset](#), se selecciona una posición mayor que el actual [LastXMLData](#) / [LastXMLDataOffset](#) (este último se mueve a la nueva posición de inicio).

CONSEJO: puede usar las propiedades [FirstXMLData](#) y [LastXMLData](#) para acceder directamente al documento XML subyacente y manipularlo en los casos en que los métodos disponibles con el objeto [AuthenticRange](#) no sean suficientes.

Errores

- 2001 El objeto de intervalo o su objeto relacionado no es válido.
- 2005 La dirección indicada para el parámetro de devolución no es válida.
- 2008 Error interno
- 2009 No se puede acceder al objeto XMLData.

Ejemplos

```
' -----  
'           VBScript  
' mostrar nombre del elemento XMLData que está seleccionado  
' -----  
  
Dim objAuthenticView  
Set objAuthenticView = objPlugin.AuthenticView  
  
Dim objXMLData  
Set objXMLData = objAuthenticView.Selection.FirstXMLData  
' la vista Authentic añade un elemento secundario "text" a los elementos  
' del documento que tienen contenido. Así que tenemos que subir  
' un elemento.  
Set objXMLData = objXMLData.Parent  
MsgBox "La selección actual selecciona el elemento " & objXMLData.Name
```

2.9.17 AuthenticRange.FirstXMLDataOffset

Propiedad: FirstXMLDataOffset as Long

Descripción

Establecer u obtener el desplazamiento de la posición del cursor dentro del elemento [FirstXMLData](#) para el principio del intervalo. Las posiciones de desplazamiento se basan en los caracteres que devuelve la propiedad [Text](#) y empiezan en la posición 0. Cuando establezca un desplazamiento nuevo, utilice -1 para establecer el desplazamiento en la última posición posible del documento. En estos casos es recomendable prestar particular atención:

- La forma textual de las entradas en cuadros combinados, casillas de verificación y controles similares pueden diferir de lo que aparece en pantalla. Aunque el desplazamiento de los datos se basa en este texto, solamente hay dos posiciones de desplazamiento válidas: una al principio y otra al final de la entrada. Si intenta establecer el desplazamiento en cualquier otra posición en el medio de la entrada, el desplazamiento se establece al final.
- La forma textual de las entidades XML pueden diferir en longitud de su representación en pantalla. El desplazamiento está basado en esta forma textual.

Si [FirstXMLData](#) / [FirstXMLDataOffset](#) selecciona una posición situada después del actual [LastXMLData](#) / [LastXMLDataOffset](#), este último se mueve hasta la posición de inicio nueva.

Errores

- 2001 El objeto de intervalo o su objeto relacionado no es válido.
2005 El desplazamiento indicado no es válido.
La dirección indicada para el parámetro de devolución no es válida.

Ejemplos

```
' -----
'           VBScript
' Seleccionar todo el texto de un elemento XMLData
' usando selección basada en XMLData y ExpandTo
' -----
Dim objAuthenticView
Set objAuthenticView = objPlugin.AuthenticView

' primero usamos las propiedades de intervalo basadas en XMLData
' para seleccionar todo el texto del primer elemento XMLData
' de la selección actual
Dim objRange
Set objRange = objAuthenticView.Selection
objRange.FirstXMLDataOffset = 0 ' empezar al principio del elemento text
objRange.LastXMLData = objRange.FirstXMLData ' seleccionar un elemento
solamente
objRange.LastXMLDataOffset = -1 ' seleccionar hasta el final

' esto mismo se puede hacer con el método ExpandTo
Dim objRange2
Set objRange2 = objAuthenticView.Selection.ExpandTo(spyAuthenticTag)

' ¿lo hemos conseguido?
If objRange.IsEqual(objRange2) Then
    objRange.Select()
Else
```

```
    MsgBox "Ooops"
End If
```

2.9.18 AuthenticRange.GetElementAttributeNames

Método: `GetElementAttributeNames (strElementName as String, out_arrAttributeNames as Variant)`

Descripción

Recupera el nombre de todos los atributos para el elemento envolvente que tiene el nombre indicado. Use pares de elementos/atributos para establecer u obtener el valor de atributo con los métodos [GetElementAttributeValue](#) y [SetElementAttributeValue](#).

Errores

- 2001 El objeto de intervalo o su objeto relacionado ya no es válido.
- 2005 El nombre de elemento indicado no es válido.
 - La dirección indicada para el parámetro de devolución no es válida.

Ejemplos

Consulte también [SetElementAttributeValue](#).

2.9.19 AuthenticRange.GetElementAttributeValue

Método: `GetElementAttributeValue (strElementName as String, strAttributeName as String) as String`

Descripción

Recuperar el valor del atributo especificado en `strAttributeName`, para el elemento identificado con `strElementName`. Si el atributo está permitido pero no tiene asignado ningún valor, se devuelve la cadena vacía. Para buscar el nombre de los atributos que admite un elemento utilice [GetElementAttributeNames](#) o [HasElementAttribute](#).

Errores

- 2001 El objeto de intervalo o su objeto relacionado ya no es válido.
- 2005 El nombre de elemento indicado no es válido.
 - El nombre de atributo indicado no es válido.
 - La dirección indicada para el parámetro de devolución no es válida.

Ejemplos

Consulte también [SetElementAttributeValue](#).

2.9.20 AuthenticRange.GetElementHierarchy

Método: `GetElementHierarchy (out_arrElementNames as Variant)`

Descripción

Recuperar el nombre de todos los elementos XML que son elementos primarios de la selección actual. Los elementos internos se enumeran antes de los elementos envolventes. Si la selección actual no está dentro de un elemento XMLData, se devuelve una lista vacía.

Los nombres de la jerarquía de elementos, junto con el objeto de intervalo, identifican de forma inequívoca los elementos XMLData del documento. A los atributos de estos elementos se puede acceder directamente con [GetElementAttributeNames](#) y sus métodos.

Errores

- 2001 El objeto de intervalo o su objeto relacionado ya no es válido.
- 2005 La dirección indicada para el parámetro de devolución no es válida.

Ejemplos

Consulte también [SetElementAttributeValue](#).

2.9.21 AuthenticRange.GetEntityNames

Método: `GetEntityNames` (*out_arrEntityNames* as Variant)

Descripción

Recuperar los nombres de todas las entidades definidas. La lista de las entidades recuperadas es independiente de la selección o ubicación actual. Use uno de estos nombres con la función [InsertEntity](#).

Errores

- 2001 El objeto de intervalo o su objeto relacionado ya no es válido.
- 2005 La dirección indicada para el parámetro de devolución no es válida.

Ejemplos

Consulte también [InsertEntity](#).

2.9.22 AuthenticRange.GetVariableValue

Método: [GetVariableValue](#) (nombre como cadena, valor como cadena)

Valor devuelto

Obtiene el valor de la variable nombrada.

Errores

- 2001 Objeto no válido.
- 2005 Parámetro no válido
- 2202 No hay contexto.
- 2204 Variable fuera de intervalo
- 2205 La variable no se puede evaluar
- 2206 La variable devuelve una secuencia

2.9.23 AuthenticRange.Goto

Método: Goto (*eKind* as SPYAuthenticElementKind, *nCount* as Long, *eFrom* as SPYAuthenticDocumentPosition) as [AuthenticRange](#)

Descripción

Configura el intervalo para apuntar al principio del elemento nº *nCount* de tipo *eKind*. La posición inicial viene definida por el parámetro *eFrom*.

Utilice valores positivos para *nCount* para navegar hasta el final del documento. Para navegar hasta el principio del documento utilice valores negativos. El método devuelve el objeto de intervalo modificado.

Errores

- 2001 El objeto de intervalo o su objeto relacionado ya no es válido.
- 2003 El destino está después del final del documento.
- 2004 El destino está antes del principio del documento.
- 2005 El tipo de elemento indicado no es válido.
 - La posición inicial indicada no es válida.
 - La dirección indicada para el parámetro de devolución no es válida.

2.9.24 AuthenticRange.GotoNext

Método: `GotoNext (eKind as SPYAuthenticElementKind) as AuthenticRange`

Descripción

Establece el intervalo en el principio del siguiente elemento de tipo `eKind`. El método devuelve el objeto de intervalo modificado.

Errores

- 2001 El objeto de intervalo o su objeto relacionado ya no es válido.
- 2003 El destino está después del final del documento.
- 2005 El tipo de elemento indicado no es válido.
La dirección indicada para el parámetro de devolución no es válida.

Ejemplos

```
' -----  
'           VBScript  
' Explorar todo el documento palabra por palabra  
' -----  
  
Dim objAuthenticView  
Set objAuthenticView = objPlugin.AuthenticView  
  
Dim objRange  
Set objRange = objAuthenticView.DocumentBegin  
Dim bEndOfDocument  
bEndOfDocument = False  
  
On Error Resume Next  
While Not bEndOfDocument  
    objRange.GotoNext(spyAuthenticWord).Select  
    If ((Err.number - vbObjecterror) = 2003) Then  
        bEndOfDocument = True  
        Err.Clear  
    ElseIf (Err.number <> 0) Then  
        Err.Raise ' error de reenvío  
    End If  
Wend
```

2.9.25 AuthenticRange.GotoNextCursorPosition

Método: [GotoNextCursorPosition \(\) as AuthenticRange](#)

Descripción

Establece el intervalo en la siguiente posición del cursor después de su posición final actual. Devuelve el objeto modificado.

Errores

- 2001 El objeto de intervalo o su objeto relacionado ya no es válido.
- 2003 El destino está después del final del documento.
- 2005 La dirección indicada para el parámetro de devolución no es válida.

2.9.26 AuthenticRange.GotoPrevious

Método: `GotoPrevious (eKind as SPYAuthenticElementKind) as AuthenticRange`

Descripción

Establece el intervalo en el principio del elemento de tipo `eKind` que está antes del principio del intervalo actual. El método devuelve el objeto de intervalo modificado.

Errores

- 2001 El objeto de intervalo o su objeto relacionado ya no es válido.
- 2004 El destino está antes del principio del documento.
- 2005 El tipo de elemento indicado no es válido.
La dirección indicada para el parámetro de devolución no es válida.

Ejemplos

```
' -----  
'           VBScript  
' Explorar todo el documento etiqueta por etiqueta  
' -----  
  
Dim objAuthenticView  
Set objAuthenticView = objPlugin.AuthenticView  
  
Dim objRange  
Set objRange = objAuthenticView.DocumentEnd  
Dim bEndOfDocument  
bBeginOfDocument = False  
  
On Error Resume Next  
While Not bBeginOfDocument  
    objRange.GotoPrevious(spyAuthenticTag).Select  
    If ((Err.number - vbObjecterror) = 2004) Then  
        bBeginOfDocument = True  
        Err.Clear  
    ElseIf (Err.number <> 0) Then  
        Err.Raise ' error de reenvío  
    End If  
Wend
```

2.9.27 AuthenticRange.GotoPreviousCursorPosition

Método: [GotoPreviousCursorPosition \(\) as AuthenticRange](#)

Descripción

Establece el intervalo en la posición del cursor situada inmediatamente antes de la posición actual. Devuelve el objeto modificado.

Errores

- 2001 El objeto de intervalo o su objeto relacionado ya no es válido.
- 2004 El destino está antes del principio del documento.
- 2005 La dirección indicada para el parámetro de devolución no es válida.

2.9.28 AuthenticRange.HasElementAttribute

Método: HasElementAttribute (*strElementName* as String, *strAttributeName* as String) as Boolean

Descripción

Comprueba si el elemento envolvente llamado *strElementName* admite el atributo indicado por *strAttributeName*.

Errores

- 2001 El objeto de intervalo o su objeto relacionado ya no es válido.
- 2005 El nombre de elemento indicado no es válido.
La dirección indicada para el parámetro de devolución no es válida.

2.9.29 AuthenticRange.InsertEntity

Método: `InsertEntity (strEntityName as String)`

Descripción

Reemplaza la selección del intervalo con la entidad indicada, que debe ser una de las entidades devueltas por [GetEntityNames](#).

Errores

- 2001 El objeto de intervalo o su objeto relacionado ya no es válido.
- 2005 Nombre de entrada desconocido.

Ejemplos

```
' -----  
' VBScript  
' Insertar la primera entidad de la lista de entidades disponibles  
' -----  
Dim objRange  
Set objRange = objPlugin.AuthenticView.Selection  
  
' primero obtenemos el nombre de todas las entidades  
' tal y como aparecen en el ayudante de entrada de XMLSpy  
Dim arrEntities  
objRange.GetEntityNames arrEntities  
  
' insertamos la primera de la lista  
If UBound(arrEntities) >= 0 Then  
    objRange.InsertEntity arrEntities(0)  
    objRange.Select()  
Else  
    MsgBox "Para este documento no hay entidades"  
End If
```

2.9.30 AuthenticRange.InsertRow

Método: `InsertRow () as Boolean`

Descripción

Si el principio del intervalo está dentro de una tabla dinámica, este método inserta una fila nueva antes de la fila actual. La selección del intervalo se modifica para apuntar al principio de la fila recién insertada. La función devuelve *true* si la operación de inserción se completa correctamente. De lo contrario devuelve *false*.

Errores

- 2001 El objeto de intervalo o su objeto relacionado ya no es válido.
- 2005 La dirección indicada para el parámetro de devolución no es válida.

Ejemplos

```
' -----  
'           VBScript  
' Insertar fila al principio de la fila dinámica actual  
' -----  
  
Dim objRange  
Set objRange = objPlugin.AuthenticView.Selection  
  
' comprobar si se puede insertar algo  
If objRange.IsInDynamicTable Then  
    objRange.InsertRow  
    ' objRange apunta al principio de la nueva fila  
    objRange.Select  
End If
```

2.9.31 AuthenticRange.IsCopyEnabled

Propiedad: `IsCopyEnabled` as Boolean (solo lectura)

Descripción

Comprueba si la operación Copiar está permitida para este intervalo.

Errores

- 2001 El objeto de intervalo o su objeto relacionado ya no es válido.
- 2005 La dirección indicada para el parámetro de devolución no es válida.

2.9.32 AuthenticRange.IsCutEnabled

Propiedad: `IsCutEnabled` as Boolean (solo lectura)

Descripción

Comprueba si la operación Cortar está permitida para este intervalo.

Errores

- 2001 El objeto de intervalo o su objeto relacionado ya no es válido.
- 2005 La dirección indicada para el parámetro de devolución no es válida.

2.9.33 AuthenticRange.IsDeleteEnabled

Propiedad: `IsDeleteEnabled` as Boolean (solo lectura)

Descripción

Comprueba si la operación Eliminar está permitida para este intervalo.

Errores

- 2001 El objeto de intervalo o su objeto relacionado ya no es válido.
- 2005 La dirección indicada para el parámetro de devolución no es válida.

2.9.34 AuthenticRange.IsEmpty

Método: `IsEmpty () as Boolean`

Descripción

Comprueba si la primera y la última posición del intervalo son iguales.

Errores

- 2001 El objeto de intervalo o su objeto relacionado ya no es válido.
- 2005 La dirección indicada para el parámetro de devolución no es válida.

2.9.35 AuthenticRange isEqual

Método: `isEqual (objCmpRange as AuthenticRange) as Boolean`

Descripción

Comprueba si el principio y el final de ambos intervalos son el mismo.

Errores

- 2001 Uno de los dos objetos de intervalo comparados no es válido.
- 2005 Dirección no válida para un parámetro de devolución.

2.9.36 AuthenticRange.IsFirstRow

Propiedad: `IsFirstRow()` as Boolean (solo lectura)

Descripción

Comprueba si el intervalo está en la primera fila de una tabla. La tabla que se consulta depende de la extensión del intervalo. Si la selección sobrepasa una sola fila de una tabla, se comprueba si esta tabla es el primer elemento de una tabla de incrustación. Consulte la documentación de los ayudantes de entrada para obtener más información.

Errores

- 2001 El objeto de intervalo o su objeto relacionado ya no es válido.
- 2005 La dirección indicada para el parámetro de devolución no es válida.

2.9.37 AuthenticRange.IsInDynamicTable

Método: `IsInDynamicTable () as Boolean`

Descripción

Comprobar si el principio del intervalo está dentro de una tabla que admite las diferentes operaciones en filas.

Errores

- 2001 El objeto de intervalo o su objeto relacionado ya no es válido.
- 2005 La dirección indicada para el parámetro de devolución no es válida.

2.9.38 AuthenticRange.IsLastRow

Propiedad: `IsLastRow()` as Boolean (solo lectura)

Descripción

Comprobar si el intervalo está en la última fila de una tabla. La tabla que se consulta depende de la extensión del intervalo. Si la selección sobrepasa una sola fila de una tabla, se comprueba si esta tabla es el último elemento de una tabla de incrustación. Consulte la documentación de los ayudantes de entrada para obtener más información.

Errores

- 2001 El objeto de intervalo o su objeto relacionado ya no es válido.
- 2005 La dirección indicada para el parámetro de devolución no es válida.

2.9.39 AuthenticRange.IsPasteEnabled

Propiedad: `IsPasteEnabled` as Boolean (solo lectura)

Descripción

Comprueba si la operación Pegar está permitida para este intervalo.

Errores

- 2001 El objeto de intervalo o su objeto relacionado ya no es válido.
- 2005 La dirección indicada para el parámetro de devolución no es válida.

2.9.40 AuthenticRange.isSelected

Propiedad: `IsSelected` as Boolean

Descripción

Devuelve true() si está presente la selección. El intervalo de selección puede estar vacío (p. ej. si solamente está el cursor).

2.9.41 AuthenticRange.IsTextStateApplied

Método: `IsTextStateApplied (i_strElementName as String) as Boolean`

Descripción

Comprueba si todo el texto seleccionado está incrustado en un elemento XML llamado `i_strElementName`. Algunos ejemplos típicos para el parámetro `i_strElementName` son "strong", "bold" y "italic".

Errores

- 2001 El objeto de intervalo o su objeto relacionado ya no es válido.
- 2005 La dirección indicada para el parámetro de devolución no es válida.

2.9.42 AuthenticRange.LastTextPosition

Propiedad: [LastTextPosition](#) as Long

Descripción

Establecer u obtener el índice de posición del texto situado más a la derecha del objeto de intervalo. Este índice siempre es mayor o igual que [FirstTextPosition](#). La indización empieza por 0 al principio del documento e incrementa con cada nueva posición que pueda ocupar el cursor de texto. Incrementar la posición de prueba 1 punto tiene el mismo efecto que la tecla a la derecha del cursor. Reducir la posición de prueba 1 punto tiene el mismo efecto que la tecla a la izquierda del cursor.

Si para *LastTextPosition* define un valor menor que el actual [FirstTextPosition](#), [FirstTextPosition](#) se establece en el nuevo *LastTextPosition*.

CONSEJO: tenga cuidado al utilizar las posiciones del cursor de texto porque es una operación costosa comparada con el posicionamiento del cursor basado en XMLData.

Errores

- 2001 El objeto de intervalo o su objeto relacionado no es válido.
- 2005 La dirección indicada para el parámetro de devolución no es válida.
- 2006 Se indicó una posición del texto situada fuera del documento.

Ejemplos

```
' -----  
'           VBScript  
' -----  
  
Dim objAuthenticView  
Set objAuthenticView = objPlugin.AuthenticView  
  
nDocStartPosition = objAuthenticView.DocumentBegin.FirstTextPosition  
nDocEndPosition = objAuthenticView.DocumentEnd.FirstTextPosition  
  
' creamos un intervalo que selecciona todo el documento  
' de manera eficaz  
Dim objRange  
' primero tenemos que obtener un objeto de intervalo (cualquiera)  
Set objRange = objAuthenticView.DocumentBegin  
objRange.FirstTextPosition = nDocStartPosition  
objRange.LastTextPosition = nDocEndPosition  
  
' comprobamos si lo hemos hecho correctamente  
If objRange.isEqual(objAuthenticView.WholeDocument) Then  
    MsgBox "Funcionó la prueba usando posicionamiento directo del cursor de texto"  
Else  
    MsgBox "Ooops!"  
End If
```

2.9.43 AuthenticRange.LastXMLData

Consulte también

Propiedad: [LastXMLData](#) as [XMLData](#)

Descripción

Establecer u obtener el último elemento XMLData del documento subyacente que esté parcial o completamente seleccionado por el intervalo. El final exacto de la selección viene definido por el atributo [LastXMLDataOffset](#).

Cuando establezca *LastXMLData* en un objeto de datos nuevo, [LastXMLDataOffset](#) se establece en la última posición del cursor dentro de este elemento. Solamente se pueden utilizar objetos XMLData que tengan una posición de cursor. Si establece *LastXMLData* / [LastXMLDataOffset](#), se selecciona una posición menor que el actual [FirstXMLData](#) / [FirstXMLDataOffset](#) (este último se mueve a la nueva posición final).

CONSEJO: puede usar las propiedades [FirstXMLData](#) y [LastXMLData](#) para acceder directamente al documento XML subyacente y manipularlo en los casos en que los métodos disponibles con el objeto [AuthenticRange](#) no sean suficientes.

Errores

- 2001 El objeto de intervalo o su objeto relacionado no es válido.
- 2005 La dirección indicada para el parámetro de devolución no es válida.
- 2008 Error interno
- 2009 No se puede acceder al objeto XMLData.

2.9.44 AuthenticRange.LastXMLDataOffset

Propiedad: [LastXMLDataOffset](#) as Long

Descripción

Establecer u obtener la posición del cursor dentro del elemento [LastXMLData](#) para el final del intervalo.

Las posiciones de desplazamiento se basan en los caracteres que devuelve la propiedad [Text](#) y empiezan en la posición 0. Cuando establezca un desplazamiento nuevo, utilice -1 para establecer el desplazamiento en la última posición posible del documento. En estos casos es recomendable prestar particular atención:

- La forma textual de las entradas en cuadros combinados, casillas de verificación y controles similares pueden diferir de lo que aparece en pantalla. Aunque el desplazamiento de los datos se basa en este texto, solamente hay dos posiciones de desplazamiento válidas: una al principio y otra al final de la entrada. Si intenta establecer el desplazamiento en cualquier otra posición en el medio de la entrada, el desplazamiento se establece al final.
- La forma textual de las entidades XML pueden diferir en longitud de su representación en pantalla. El desplazamiento está basado en esta forma textual.

Si [LastXMLData](#) / [LastXMLDataOffset](#) selecciona una posición situada antes del actual [FirstXMLData](#) / [FirstXMLDataOffset](#), este último se mueve hasta la posición final nueva.

Errores

- 2001 El objeto de intervalo o su objeto relacionado no es válido.
2005 El desplazamiento indicado no es válido.
La dirección indicada para el parámetro de devolución no es válida.

Ejemplos

```
' -----
'          VBScript
' Seleccionar todo el texto de un elemento XMLData
' usando selección basada en XMLData y ExpandTo
' -----
Dim objAuthenticView
Set objAuthenticView = objPlugin.AuthenticView

' primero usamos las propiedades de intervalo basadas en XMLData
' para seleccionar todo el texto del primer elemento XMLData
' de la selección actual
Dim objRange
Set objRange = objAuthenticView.Selection
objRange.FirstXMLDataOffset = 0 ' empezar al principio del elemento text
objRange.LastXMLData = objRange.FirstXMLData ' seleccionar un elemento
solamente
objRange.LastXMLDataOffset = -1 ' seleccionar hasta el final

' esto mismo se puede hacer con el método ExpandTo
Dim objRange2
Set objRange2 = objAuthenticView.Selection.ExpandTo(spyAuthenticTag)

' ¿lo hemos conseguido?
```

```
If objRange.AreEqual(objRange2) Then  
    objRange.Select()  
Else  
    MsgBox "Ooops"  
End If
```

2.9.45 AuthenticRange.MoveBegin

Método: `MoveBegin (eKind as SPYAuthenticElementKind, nCount as Long) as AuthenticRange`

Descripción

Mover el inicio del intervalo al principio del elemento nº *nCount* de tipo *eKind*. El recuento comienza por el inicio actual del objeto de intervalo.

Utilice números positivos para *nCount* para desplazarse hacia el final del documento y negativos para desplazarse hacia el principio. El final del intervalo no se mueve, a no ser que el nuevo inicio sea mayor. En este caso el final se mueve al nuevo inicio. El método devuelve el objeto de intervalo modificado.

Errores

- 2001 El objeto de intervalo o su objeto relacionado ya no es válido.
- 2003 El destino está después del final del documento.
- 2004 El destino está antes del principio del documento.
- 2005 El tipo de elemento indicado no es válido.
La dirección indicada para el parámetro de devolución no es válida.

2.9.46 AuthenticRange.MoveEnd

Método: MoveEnd (*eKind* as SPYAuthenticElementKind, *nCount* as Long) as [AuthenticRange](#)

Descripción

Mover el final del intervalo al principio del elemento nº *nCount* de tipo *eKind*. El recuento comienza por el final actual del objeto de intervalo.

Utilice números positivos para *nCount* para desplazarse hacia el final del documento y negativos para desplazarse hacia el principio. El inicio del intervalo no se mueve, a no ser que el nuevo final sea menor. En este caso el inicio se mueve al nuevo final. El método devuelve el objeto de intervalo modificado.

Errores

- 2001 El objeto de intervalo o su objeto relacionado ya no es válido.
- 2003 El destino está después del final del documento.
- 2004 El destino está antes del principio del documento.
- 2005 El tipo de elemento indicado no es válido.
La dirección indicada para el parámetro de devolución no es válida.

2.9.47 AuthenticRange.MoveRowDown

Método: `MoveRowDown () as Boolean`

Descripción

Si el principio del intervalo está dentro de una tabla dinámica y selecciona una fila que no es la última fila de esta tabla, este método intercambia esta fila con la fila situada justo debajo. La selección del intervalo se mueve con la fila, pero sigue en el mismo lugar. La función devuelve *true* si la operación finaliza correctamente. De lo contrario devuelve *false*.

Errores

- 2001 El objeto de intervalo o su objeto relacionado ya no es válido.
- 2005 La dirección indicada para el parámetro de devolución no es válida.

2.9.48 AuthenticRange.MoveRowUp

Método: `MoveRowUp () as Boolean`

Descripción

Si el principio del intervalo está dentro de una tabla dinámica y selecciona una fila que no es la primera fila de esta tabla, este método intercambia esta fila con la fila situada encima. La selección del intervalo se mueve con la fila, pero sigue en el mismo lugar. La función devuelve `true` si la operación finaliza correctamente. De lo contrario devuelve `false`.

Errores

- 2001 El objeto de intervalo o su objeto relacionado ya no es válido.
- 2005 La dirección indicada para el parámetro de devolución no es válida.

Ejemplos

Consulte también [Ejemplo: ordenar una tabla](#).

2.9.49 AuthenticRange.Parent

Consulte también

Propiedad: Parent as [AuthenticView](#) (solo lectura)

Descripción

Acceder a la vista propietaria de este objeto de intervalo.

Errores

- 2001 El objeto de intervalo o su objeto relacionado ya no es válido.
- 2005 La dirección indicada para el parámetro de devolución no es válida.

2.9.50 AuthenticRange.Paste

Método: Paste () as Boolean

Descripción

Devuelve *False* si el búfer de copia/pega está vacío o si su contenido no puede reemplazar la selección actual.

De lo contrario, elimina la selección actual, inserta el contenido del búfer de copia/pega y devuelve *True*.

Errores

- 2001 El objeto de intervalo o su objeto relacionado ya no es válido.
- 2005 La dirección indicada para el parámetro de devolución no es válida.

2.9.51 AuthenticRange.PerformAction

Método: PerformAction (*eAction* as SPYAuthenticActions, *strElementName* as String) as Boolean

Descripción

PerformAction y sus métodos ofrecen acceso a las funciones de los ayudantes de entrada de Authentic. Esta función permite modificar el contenido del documento de forma sencilla y coherente sin necesidad de saber dónde tienen lugar los cambios exactamente. El principio del objeto de intervalo se usa para encontrar la siguiente ubicación válida donde se puede llevar a cabo la acción indicada. Si dicha ubicación no se encuentra, el método devuelve False. De lo contrario, el documento se modifica y el intervalo apunta al principio de la modificación.

CONSEJO: utilice [CanPerformActionWith](#) para averiguar el nombre de los elementos que se pueden pasar como segundo parámetro.

Errores

- 2001 El objeto de intervalo o su objeto relacionado ya no es válido.
- 2005 La dirección indicada para el parámetro de devolución no es válida.
- 2007 La acción indicada no es válida.

Ejemplos

```
' -----
'           VBScript
' Insertar el elemento más interior
' -----
Dim objRange
Set objRange = objPlugin.AuthenticView.Selection

' determinamos qué elementos se pueden insertar en la posición actual
Dim arrElements()
objRange.CanPerformActionWith spyAuthenticInsertBefore, arrElements

' insertamos el primer elemento (es decir, el más interior)
If UBound(arrElements) >= 0 Then
    objRange.PerformAction spyAuthenticInsertBefore, arrElements(0)
    ' objRange ahora apunta al principio del elemento insertado
    ' establecemos un valor y una posición predeterminados en su final
    objRange.Text = "Hello"
    objRange.ExpandTo(spyAuthenticTag).CollapseToEnd().Select
Else
    MsgBox "En la posición actual no se puede insertar ningún elemento"
End If
```

2.9.52 AuthenticRange.Select

Método: [Select \(\)](#)

Descripción

Convierte el intervalo en la selección actual de la interfaz del usuario. Con '*objRange.Parent.Selection = objRange*' puede conseguir el mismo resultado.

Errores

2001 El objeto de intervalo o su objeto relacionado ya no es válido.

Ejemplos

```
' establecer selección actual en final del documento
objPlugin,objAuthenticView.DocumentEnd.Select()
```

2.9.53 AuthenticRange.SelectNext

Método: `SelectNext (eKind as SPYAuthenticElementKind) as AuthenticRange`

Descripción

Selecciona el elemento de tipo *eKind* situado tras el final actual del intervalo. El método devuelve el objeto de intervalo modificado.

Errores

- 2001 El objeto de intervalo o su objeto relacionado ya no es válido.
- 2003 El destino está después del final del documento.
- 2005 El tipo de elemento indicado no es válido.
La dirección indicada para el parámetro de devolución no es válida.

Ejemplos

```
' -----  
' VBScript  
' Explorar todo el documento palabra por palabra  
' -----  
  
Dim objAuthenticView  
Set objAuthenticView = objPlugin.AuthenticView  
  
Dim objRange  
Set objRange = objAuthenticView.DocumentBegin  
Dim bEndOfDocument  
bEndOfDocument = False  
  
On Error Resume Next  
While Not bEndOfDocument  
    objRange.SelectNext(spyAuthenticWord).Select  
    If ((Err.number - vbObjecterror) = 2003) Then  
        bEndOfDocument = True  
        Err.Clear  
    ElseIf (Err.number <> 0) Then  
        Err.Raise ' error de reenvío  
    End If  
Wend
```

2.9.54 AuthenticRange.SelectPrevious

Método: [GotoPrevious \(eKind as SPYAuthenticElementKind\) as AuthenticRange](#)

Descripción

Selecciona el elemento de tipo *eKind* situado antes del inicio actual del intervalo. El método devuelve el objeto de intervalo modificado.

Errores

- 2001 El objeto de intervalo o su objeto relacionado ya no es válido.
- 2004 El destino está antes del principio del documento.
- 2005 El tipo de elemento indicado no es válido.
La dirección indicada para el parámetro de devolución no es válida.

Ejemplos

```
' -----  
'           VBScript  
' Explorar todo el documento etiqueta por etiqueta  
' -----  
  
Dim objAuthenticView  
Set objAuthenticView = objPlugin.AuthenticView  
  
Dim objRange  
Set objRange = objAuthenticView.DocumentEnd  
Dim bEndOfDocument  
bBeginOfDocument = False  
  
On Error Resume Next  
While Not bBeginOfDocument  
    objRange.SelectPrevious(spyAuthenticTag).Select  
    If ((Err.number - vbObjecterror) = 2004) Then  
        bBeginOfDocument = True  
        Err.Clear  
    ElseIf (Err.number <> 0) Then  
        Err.Raise ' error de reenvío  
    End If  
Wend
```

2.9.55 AuthenticRange.SetElementAttributeValue

Método: SetElementAttributeValue (*strElementName* as String, *strAttributeName* as String, *strAttributeValue* as String)

Descripción

Recuperar el valor del atributo especificado en *strAttributeName* para el elemento identificado por *strElementName*. Si el atributo es compatible pero no tiene asignado ningún valor, se devuelve una cadena vacía. Para averiguar el nombre de los atributos compatibles con un elemento utilice [GetElementAttributeNames](#) o [HasElementAttribute](#).

Errores

- 2001 El objeto de intervalo o su objeto relacionado ya no es válido.
- 2005 El nombre de elemento indicado no es válido.
 - El nombre de atributo indicado no es válido.
 - Se indicó un valor de atributo no válido.

Ejemplos

```
' -----
'           VBScript
' Obtener y establecer atributos del elemento
' -----

Dim objRange
Set objRange = objPlugin.AuthenticView.Selection

' primero averiguamos todos los elementos situados al principio del intervalo
Dim arrElements
objRange.GetElementHierarchy arrElements

If IsArray(arrElements) Then
    If UBound(arrElements) >= 0 Then
        ' usamos el elemento de nivel superior y averiguamos sus atributos
        válidos
        Dim arrAttrs()
        objRange.GetElementAttributeNames arrElements(0), arrAttrs

        If UBound(arrAttrs) >= 0 Then
            ' recuperamos el valor actual del primer atributo válido
            Dim strAttrVal
            strAttrVal = objRange.GetElementAttributeValue
            (arrElements(0), arrAttrs(0))
            msgbox "el valor actual de " & arrElements(0) & "://" &
            arrAttrs(0) & " es: " & strAttrVal

            ' cambiamos este valo y lo volvemos a leer
            strAttrVal = "Hello"
            objRange.SetElementAttributeValue arrElements(0),
            arrAttrs(0), strAttrVal
            strAttrVal = objRange.GetElementAttributeValue
            (arrElements(0), arrAttrs(0))
            msgbox "el nuevo valor de " & arrElements(0) & "://" &
            arrAttrs(0) & " es: " & strAttrVal
        End If
    End If
End If
```

End If

2.9.56 AuthenticRange.SetFromRange

Método: `SetFromRange (objSrcRange as AuthenticRange)`

Descripción

Establece el objeto de intervalo en las mismas posiciones de inicio y fin que *objSrcRange*.

Errores

- 2001 Uno de los dos objetos de intervalo no es válido.
- 2005 Se indicó un objeto null como objeto de origen.

2.9.57 AuthenticRange.SetValue

Método: `SetValue(nombre como cadena, valor como cadena)`

Valor devuelto

Establece el valor de la variable nombrada.

Errores

- 2001 Objeto no válido.
- 2002 Falta el nodo de contexto.
- 2204 Variable fuera de intervalo
- 2205 La variable no se puede evaluar
- 2206 La variable devuelve una secuencia
- 2207 Variable de solo lectura
- 2208 No se permiten cambios

2.9.58 AuthenticRange.Text

Propiedad: `Text` as `String`

Descripción

Establecer u obtener el contenido textual seleccionado por el objeto de intervalo.

El número de caracteres recuperados no es necesariamente idéntico porque hay posiciones del cursor de texto entre el principio y el final del intervalo seleccionado. La mayoría de los elementos del documento admiten una posición final del cursor diferente a la posición inicial del cursor del elemento siguiente. Las listas desplegables conservan una sola posición del cursor, pero pueden seleccionar cadenas de cualquier longitud. En el caso de los botones de opción y de las casillas de verificación, el valor de la propiedad `Text` almacena la cadena del elemento XML correspondiente.

Si el intervalo selecciona más de un elemento, el texto es la concatenación de los textos. Las entidades XML se expanden para que '`&`' se espere como '`&`'.

Al asignarse el texto a la cadena vacía, no se elimina ningún elemento XML. Utilice [Cut](#), [Delete](#) o [PerformAction](#) en su lugar.

Errores

- 2001 El objeto de intervalo o su objeto relacionado ya no es válido.
- 2005 Dirección no válida para un parámetro de devolución.

2.10 AuthenticSelection

Propiedades

[Start](#)

[StartTextPosition](#)

[End](#)

[EndTextPosition](#)

2.10.1 AuthenticSelection.End

Declaración End as [XMLData](#)

Descripción

Elemento XML donde finaliza la selección actual.

2.10.2 AuthenticSelection.EndTextPosition

Declaración EndTextPosition as long

Descripción

Posición en [Authentic.End](#) donde finaliza la selección.

2.10.3 AuthenticSelection.Start

Declaración Start as [XMLData](#)

Descripción

Elemento XML donde comienza la selección actual.

2.10.4 AuthenticSelection.StartTextPosition

Declaración StartTextPosition as long

Descripción

Posición en [Authentic.Start](#) donde comienza la selección.

2.11 AuthenticToolbarButton

Métodos

Propiedades

[CommandID](#)

2.11.1 AuthenticToolbarButton.CommandID

Declaración CommandID as [SPYAuthenticCommand](#)

Descripción

El CommandId para el botón de la barra de herramientas.

2.12 AuthenticToolbarButtons

Métodos

[Item](#)

[NewButton](#)

[NewCustomButton](#)

[NewSeparator](#)

[Remove](#)

Propiedades

[Count](#)

2.12.1 AuthenticToolbarButtons.Count

Declaración Count as long

Descripción

Obtener el número de botones que hay en la barra de herramientas.

Solo lectura.

2.12.2 AuthenticToolbarButtons.Item

Declaración Item(n as long) as [AuthenticToolbarButton](#)

Descripción

Obtener el botón nº n de la barra herramientas actual. n empieza por 1.

Ejemplo

Consulte el ejemplo disponible en [AuthenticToolbarButtons.NewButton](#)

2.12.3 AuthenticToolbarButtons.NewButton

Declaración NewButton(*nPosition* as long, *nCommandId* as [SPYAuthenticCommand](#))

Descripción

Inserta un botón nuevo para el comando *nCommandId* en la posición *nPosition* de la barra de herramientas. *nPosition* empieza por 1.

Ejemplo

Añadir una barra de herramientas nueva, alinearla en la parte inferior y añadir botones nuevos

```
objPlugIn.ToolbarRows.NewReader(3)                                // añadir una barra de
                                                               // herramientas nueva (fila 3)

set ToolbarRow = objPlugIn.ToolbarRows.Item(3)
set Buttons = ToolbarRow.Buttons
ToolbarRow.Alignment = 2                                         // alinear la barra de
                                                               // herramientas en la parte
                                                               // inferior

Buttons.NewButton    1, 2                                     // agregar el botón Imprimir
Buttons.NewButton    1, 3                                     // agregar el botón Vista
                                                               // previa de impresión
Buttons.NewSeparator 2                                       // agregar separador
Buttons.NewButton    1, 4                                     // agregar el botón Validar
```

Cuando se llama a [StartEditing](#) o [ReloadToolbars](#), se usa la configuración modificada de la barra de herramientas.

2.12.4 AuthenticToolbarButtons.NewCustomButton

Declaración NewCustomButton(*nPosition* as long, *strName* as String, *strTooltip* as String, *strBitmapURL* as String)

Descripción

Inserta un botón personalizado nuevo llamado strName en la posición nPosition de la barra de herramientas. nPosition empieza por 1.
strTooltip se usa como texto de información rápida.

strBitmapURL es la ubicación del mapa de bits que debe utilizarse para el nuevo botón. La ruta de acceso es relativa a la ruta definida con la propiedad [TextStateBmpURL](#).

Ejemplo

Imagine que insertó un botón personalizado llamado MiFunción en las barras de herramientas. El siguiente controlador de eventos para [doceditcommand](#) demuestra la reacción que se produce cuando el usuario hace clic en el botón y cómo se puede configurar su estado habilitado / deshabilitado.

```
<SCRIPT LANGUAGE=javascript FOR=objPlugIn EVENT=doceditcommand>
  // event.type recibe el valor "command" si el usuario hizo clic en el botón
  if(objPlugIn.event.type == "command")
  {
    if(objPlugIn.event.srcElement.Name == "MiFunción")
      window.alert("Acaba de pulsar el botón personalizado.");
  }

  // event.type recibe el valor "update" si se debe establecer el estado del
  // botón
  if(objPlugIn.event.type == "update")
  {
    if(objPlugIn.event.srcElement.Name == "MiFunción")
    {
      // habilitamos el botón si hay un solo elemento seleccionado

      if(objPlugIn.CurrentSelection.Start.IsSameNode(objPlugIn.CurrentSelection.End))
        objPlugIn.event.returnValue = 1;
      else
        objPlugIn.event.returnValue = 0;

      objPlugIn.event.cancelBubble = true;
    }
  }
</SCRIPT>
```

2.12.5 AuthenticToolbarButtons.NewSeparator

Declaración NewSeparator(*nPosition* as long)

Descripción

Inserta un separador en la posición *nPosition* de la barra de herramientas. *nPosition* empieza por 1.

Ejemplo

Consulte el ejemplo disponible en [AuthenticToolbarButtons.NewButton](#)

2.12.6 AuthenticToolbarButtons.Remove

Declaración Remove(*nPosition* as long)

Descripción

Quita el botón o el separador de la posición *nPosition* de la barra de herramientas. *nPosition* empieza por 1.

2.13 AuthenticToolbarRow

Métodos

[Alignment](#)

Propiedades

[Buttons](#)

2.13.1 AuthenticToolbarRowAlignment

Declaración Alignment(**nAlign** as [SPYAuthenticToolbarAlignment](#))

Descripción

Obtiene o establece la alineación de la barra de herramientas dentro del complemento.

Ejemplo

Alinear todas las barras de herramientas en la parte inferior:

```
for each ToolbarRow in objPlugin.ToolbarRows
    ToolbarRow.Alignment = 2
next
```

2.13.2 AuthenticToolbarRowButtons

Declaración Buttons as [AuthenticToolbarButtons](#)

Descripción

Obtener todos los botones de la barra de herramientas.

Ejemplo

Consulte el ejemplo disponible en [AuthenticToolbarButtons.NewButton](#)

2.14 AuthenticToolbarRows

Métodos

[Item](#)

[RemoveRow](#)

[NewRow](#)

Propiedades

[Count](#)

2.14.1 AuthenticToolbarRows.Count

Declaración: Count as long

Descripción

Obtener el número de barras de herramientas definidas.

Solo lectura.

2.14.2 AuthenticToolbarRows.Item

Declaración Item(*nPosition* as long) as [AuthenticToolbarRow](#)

Descripción

Obtener la barra de herramientas de la posición *nPosition*. *nPosition* empieza por 1
Solo lectura.

Ejemplo

Consulte el ejemplo disponible en [AuthenticToolbarButtons.NewButton](#)

2.14.3 AuthenticToolbarRows.RemoveRow

Declaración RemoveRow(*nPosition* as long)

Descripción

Quitar la barra de herramientas de la posición *nPosition* de la interfaz del usuario. *nPosition* empieza por 1.

2.14.4 AuthenticToolbarRows.NewRow

Declaración `NewRow(nPosition as long)`

Descripción

Crea e inserta una fila nueva de barra de herramientas. nPosition empieza por 1.

Ejemplo

Consulte el ejemplo disponible en [AuthenticToolbarButtons.NewButton](#)

2.15 AuthenticView

Propiedades	Métodos	Eventos
Application	Goto	
DocumentBegin	Print	
DocumentEnd	Redo	
MarkupVisibility	Undo	
Parent	UpdateXMLInstanceEntities	
Selection		
WholeDocument		

Descripción

AuthenticView y su objeto secundario [AuthenticRange](#) ofrecen una interfaz para la vista Authentic que permite modificar de forma coherente y sencilla el contenido de los documentos. Si hay métodos y propiedades repetidos del objeto Authentic, las funciones de modificación de contenido del objeto Authentic se convierten en obsoletas. De hecho, se recomienda usar la nueva interfaz AuthenticView.

AuthenticView ofrece acceso a funciones concretas como la función de impresión, el búfer de las operaciones Deshacer y la selección o posición actual del cursor.

AuthenticView utiliza objetos de tipo [AuthenticRange](#) para facilitar la navegación dentro del documento y para permitir una selección flexible de los elementos de texto lógicos. Use las propiedades [DocumentBegin](#), [DocumentEnd](#) y [WholeDocument](#) para selecciones sencillas y el método [Goto](#) para selecciones más complejas. Para navegar por un intervalo concreto del documento, consulte las propiedades y los métodos del objeto [AuthenticRange](#).

2.15.1 Eventos

OnBeforeCopy

Evento: `OnBeforeCopy()` as Boolean

Entorno de scripting de XMLSpy: VBScript:

```
Function On_AuthenticBeforeCopy()
    ' On_AuthenticBeforeCopy = False    ' para deshabilitar la
    operación
End Function
```

Entorno de scripting de XMLSpy: JScript:

```
function On_AuthenticBeforeCopy()
{
    // devolver false; /* para deshabilitar la operación */
}
```

Complemento de XMLSpy para entornos IDE:

```
IXMLSpyPlugIn.OnEvent (21, ...) // nEventId = 21
```

Descripción

Este evento se desencadena antes de que se lleve a cabo una operación Copiar en el documento. Devolver *True* (o nada) para permitir la operación Copiar. Devolver *False* para deshabilitar la copia.

OnBeforeCut

Evento: `OnBeforeCut()` as Boolean

Entorno de scripting de XMLSpy: VBScript:

```
Function On_AuthenticBeforeCut()
    ' On_AuthenticBeforeCut = False    ' para deshabilitar la
    operación
End Function
```

Entorno de scripting de XMLSpy: JScript:

```
function On_AuthenticBeforeCut()
{
    // devolver false; /* para deshabilitar la operación */
}
```

Complemento de XMLSpy para entornos IDE:

```
IXMLSpyPlugIn.OnEvent (20, ...) // nEventId = 20
```

Descripción

Este evento se desencadena antes de que se lleve a cabo una operación Cortar en el documento. Devolver *True* (o nada) para permitir la operación Cortar. Devolver *False* para deshabilitar la operación.

OnBeforeDelete

Evento: `OnBeforeDelete()` as Boolean

Entorno de scripting de XMLSpy: VBScript:

```
Function On_AuthenticBeforeDelete()
    ' On_AuthenticBeforeDelete = False      ' para deshabilitar la
    operación
End Function
```

Entorno de scripting de XMLSpy: JScript:

```
function On_AuthenticBeforeDelete()
{
    // devolver false; /* para deshabilitar la operación */
}
```

Complemento de XMLSpy para entornos IDE:

```
IXMLSpyPlugIn.OnEvent (22, ...) // nEventId = 22
```

Descripción

Este evento se desencadena antes de que se lleve a cabo una operación Eliminar en el documento. Devolver *True* (o nada) para permitir la operación Eliminar. Devolver *False* para deshabilitar la operación.

OnBeforeDrop

Evento: `OnBeforeDrop (i_nXPos as Long, i_nYPos as Long, i_ipRange as AuthenticRange, i_ipData as cancelBoolean)`

Entorno de scripting de XMLSpy: VBScript:

```
Function On_AuthenticBeforeDrop(nXPos, nYPos, objRange, objData)
    ' On_AuthenticBeforeDrop = False      ' para deshabilitar la
    operación
End Function
```

Entorno de scripting de XMLSpy: JScript:

```
function On_AuthenticBeforeDrop(nXPos, nYPos, objRange, objData)
{
    // devolver false; /* para deshabilitar la operación */
}
```

Complemento de XMLSpy para entornos IDE:

```
IXMLSpyPlugIn.OnEvent (11, ...) // nEventId = 11
```

Descripción

Este evento se desencadena cada vez que arrastra y coloca un objeto dentro de la ventana de la aplicación. Toda la información relacionada con el evento se pasa en forma de parámetros.

Los dos primeros parámetros especifican la posición del mouse cuando ocurre el evento. El parámetro *objRange* pasa un objeto de intervalo que selecciona el elemento XML situado en la posición del mouse. El valor de este parámetro puede ser NULL. Compruébelo antes de acceder

al objeto de intervalo. El parámetro *objData* permite acceder a la información disponible sobre el objeto que se está arrastrando.

Devolver *False* para cancelar la operación Colocar. Devolver *True* (o nada) para continuar con la operación.

Ejemplos

```
' -----  
' Fragmento de código VB: conectarse a eventos a nivel de objeto  
' -----  
'  
' acceder a XMLSpy (sin comprobar si hay errores)  
Dim objSpy As XMLSpyLib.Application  
Set objSpy = GetObject("", "XMLSpy.Application")  
  
' esta es la rutina de devolución de llamada del evento  
' conectada al evento OnBeforeDrop del objeto objView  
Private Function objView_OnBeforeDrop(ByVal i_nXPos As Long, ByVal i_nYPos As  
Long,  
                                      ByVal i_ipRange As IAuthenticRange,  
                                      ByVal i_ipData As  
IAuthenticDataTransfer) As Boolean  
  
    If (Not i_ipRange Is Nothing) Then  
        MsgBox ("Dropping on content is prohibited");  
        Return False;  
    Else  
        Return True;  
    End If  
End Function  
  
' use la palabra clave de VBA WithEvents para conectarse al evento a nivel de  
' objeto  
Dim WithEvents objView As XMLSpyLib.AuthenticView  
Set objView = objSpy.ActiveDocument.AuthenticView  
  
' continuar con más código ...  
' y servir el bucle de mensajes
```

OnBeforePaste

Evento: *OnBeforePaste* (*objData* as Variant, *strType* as String) as Boolean

Entorno de scripting de XMLSpy: VBScript:

```
Function On_AuthenticBeforePaste(objData, strType)  
    ' On_AuthenticBeforePaste = False ' para deshabilitar la  
    ' operación  
End Function
```

Entorno de scripting de XMLSpy: JScript:

```
function On_AuthenticBeforePaste(objData, strType)  
{  
    // devolver false; /* para deshabilitar la operación */  
}
```

Complemento de XMLSpy para entornos IDE:

```
IXMLSpyPlugIn.OnEvent (19, ...) // nEventId = 19
```

Descripción

Este evento se desencadena antes de que se lleve a cabo la operación Pegar en el documento. El parámetro *strType* es "TEXT", "UNICODETEXT" o "IUNKNOWN". En los dos primeros casos *objData* contiene una representación de cadena del objeto que se pegará. En el último, *objData* contiene un puntero que apunta a una interfaz IUnknown de COM.

Devolver *True* (o nada) para permitir la operación Pegar. Devolver *False* para deshabilitar la operación.

OnBeforeSave

Evento: `OnBeforeSave (SaveAs flag) as Boolean`

Descripción: `OnBeforeSave` da la oportunidad de, por ejemplo, advertir al usuario antes de sobrescribir el documento XML o de convertir el documento en un documento de solo lectura cuando no se cumplan ciertos requisitos. El evento se desencadenará antes de que aparezca el cuadro de diálogo del archivo. **Nota:** el evento se desencadena cuando se guarda el documento XML y no cuando se guarda el diseño SPS en StyleVision.

OnDragOver

Evento: `OnDragOver (nXPos as Long, nYPos as Long, eMouseEvent as SPYMouseEvent, objRange as AuthenticRange, objData as AuthenticDataTransfer) as Boolean`

Entorno de scripting de XMLSpy: VBScript:

```
Function On_AuthenticDragOver(nXPos, nYPos, eMouseEvent, objRange,
objData)
    ' On_AuthenticDragOver = False ' para deshabilitar la operación
End Function
```

Entorno de scripting de XMLSpy: JScript:

```
function On_AuthenticDragOver(nXPos, nYPos, eMouseEvent, objRange,
objData)
{
    // devolver false; /* para deshabilitar la operación */
}
```

Complemento de XMLSpy para entornos IDE:

```
IXMLSpyPlugIn.OnEvent (10, ...) // nEventId = 10
```

Descripción

Este evento se desencadena cada vez que se arrastra un objeto desde fuera de la vista Authentic por encima de la ventana de la aplicación. Toda la información relacionada con el evento se pasa en forma de parámetros.

Los tres primeros parámetros especifican la posición del mouse, el estado del botón del mouse y el estado de las claves virtuales en el momento en que tiene lugar el evento. El parámetro

objRange pasa un objeto de intervalo que selecciona el elemento XML situado bajo la posición del mouse. El valor de este parámetro puede ser NULL. El parámetro *objData* permite acceder a la información disponible sobre el objeto que se está arrastrando.

Devolver *False* para cancelar la operación de arrastrar. Devolver *True* (o nada) para continuar con la operación.

Ejemplos

```
' -----  
' fragmento de código VB: conectarse a eventos a nivel de objeto  
' -----  
  
' acceder a XMLSpy (sin buscar errores)  
Dim objSpy As XMLSpyLib.Application  
Set objSpy = GetObject("", "XMLSpy.Application")  
  
' esta es la rutina de devolución de llamada  
' conectada al evento OnDragOver del objeto objView  
Private Function objView_OnDragOver(ByVal i_nXPos As Long, ByVal i_nYPos As  
Long,  
                                     ByVal i_eMouseEvent As SPYMouseEvent,  
                                     ByVal i_ipRange As IAuthenticRange,  
                                     ByVal i_ipData As IAuthenticDataTransfer)  
As Boolean  
  
    If (((i_eMouseEvent And spyShiftKeyDownMask) <> 0) And  
        (Not i_ipRange Is Nothing)) Then  
        MsgBox ("Floating over element " &  
i_ipRange.FirstXMLData.Parent.Name);  
    End If  
  
    Return True;  
End Function  
  
' use la palabra clave de VBA WithEvents para conectarse al evento a nivel de  
' objeto  
Dim WithEvents objView As XMLSpyLib.AuthenticView  
Set objView = objSpy.ActiveDocument.AuthenticView  
  
' continuar con más código ...  
' y servir el bucle de mensajes
```

OnKeyboardEvent

Evento: *OnKeyboardEvent* (*eKeyEvent* as SPYKeyEvent, *nKeyCode* as Long, *nVirtualKeyStatus* as Long) as Boolean

Entorno de scripting de XMLSpy: VBScript:

```
Function On_AuthenticKeyboardEvent(eKeyEvent, nKeyCode,  
nVirtualKeyStatus)  
    ' On_AuthenticKeyboardEvent = True    ' to cancel bubbling of  
    event  
End Function
```

Entorno de scripting de XMLSpy: JScript:

```

function On_AuthenticKeyboardEvent( eKeyEvent, nKeyCode,
nVirtualKeyStatus)
{
    // devolver false; /* to cancel bubbling of event */
}

```

Complemento de XMLSpy para entornos IDE:

```
IXMLSpyPlugIn.OnEvent (30, ...) // nEventId = 30
```

Descripción

Este evento se desencadena para los mensajes de Windows *WM_KEYDOWN*, *WM_KEYUP* y *WM_CHAR*.

El tipo de mensaje está disponible en el parámetro *eKeyEvent*. El estado de las teclas virtuales está en el parámetro *nVirtualKeyStatus*. Utilice las máscaras de bits definidas en el tipo de datos de enumeración *SPYVirtualKeyMask* para probar las distintas claves y sus combinaciones.

NOTA: estos eventos del entorno de scripting y del complemento de Altova XMLSpy para entornos IDE son compatibles pero se vuelven obsoletos con este evento:

```

On_AuthenticKeyUp()           IXMLSpyPlugIn.OnEvent (13, ...)   //
nEventId = 13
On_AuthenticKeyDown()         IXMLSpyPlugIn.OnEvent (12, ...)   //
nEventId = 12
On_AuthenticKeyPressed()     IXMLSpyPlugIn.OnEvent (14, ...)   //
nEventId = 14

```

Ejemplos

```

' -----
' fragmento de código VB: conectarse a eventos a nivel de objeto
' -----
' acceder a XMLSpy (sin buscar errores)
Dim objSpy As XMLSpyLib.Application
Set objSpy = GetObject("", "XMLSpy.Application")

' esta es la rutina de devolución de llamada
' conectada al evento OnKeyboard del objeto objView
Private Function objView_OnKeyboardEvent(ByVal i_keyEvent As Long, ByVal
io_pnKeyCode As Long, ByVal i_nVirtualKeyStatus As Long) As Boolean
    If ((i_keyEvent = XMLSpyLib.spyKeyUp) And ((i_nVirtualKeyStatus And
XMLSpyLib.spyCtrlKeyMask) <> 0)) Then
        MsgBox ("Ctrl " & io_pnKeyCode & " pressed")
        objView_OnKeyboardEvent = True
    Else
        objView_OnKeyboardEvent = False
    End If
End Function

' use la palabra clave de VBA WithEvents para conectarse al evento a nivel de
' objeto
Dim WithEvents objView As XMLSpyLib.AuthenticView
Set objView = objSpy.ActiveDocument.AuthenticView

' continuar con más código ...
' y servir el bucle de mensajes

```

OnLoad

Evento: OnLoad ()

Descripción: OnLoad se puede usar para, por ejemplo, restringir alguna función de la vista Authentic.

Ejemplo

```
function On_AuthenticLoad( )
{
    // Deshabilitamos todos los ayudantes de entrada para evitar que el
    // usuario manipule la estructura XML
    AuthenticView.DisableElementEntryHelper();
    AuthenticView.DisableAttributeEntryHelper();

    // Y deshabilitamos los botones de marcado por el mismo motivo
    AuthenticView.SetToolbarButtonState( 'AuthenticMarkupSmall',
    authenticToolbarButtonDisabled );
    AuthenticView.SetToolbarButtonState( 'AuthenticMarkupLarge',
    authenticToolbarButtonDisabled );
    AuthenticView.SetToolbarButtonState( 'AuthenticMarkupMixed',
    authenticToolbarButtonDisabled );
}
```

En este ejemplo el estado de los botones de la barra de herramientas **Marcado pequeño**, **Marcado mixto**, **Marcado pequeño** se manipulan con ayuda de los identificadores de botón. Aquí puede ver una [lista completa](#).

OnMouseEvent

Evento: OnMouseEvent (nXPos as Long, nYPos as Long, eMouseEvent as SPYMouseEvent, objRange as AuthenticRange) as Boolean

Entorno de scripting de XMLSpy: VBScript:

```
Function On_AuthenticMouseEvent( nXPos, nYPos, eMouseEvent, objRange )
    ' On_AuthenticMouseEvent = True    ' para cancelar la propagación
    del evento
End Function
```

Entorno de scripting de XMLSpy: JScript:

```
function On_AuthenticMouseEvent( nXPos, nYPos, eMouseEvent, objRange )
{
    // devolver false; /* para cancelar la propagación del evento
*/ }
```

Complemento de XMLSpy para entornos IDE:

```
IXMLSpyPlugIn.OnEvent (31, ...) // nEventId = 31
```

Descripción

Este evento se desencadena cada vez que se mueve el mouse y con cada mensaje de Windows para los botones del mouse.

El tipo de mensaje y el estado del botón del mouse está disponible en el parámetro `eMouseEvent`. Use las máscaras de bits definidas en el tipo de datos de enumeración `SPYMouseEvent` para probar los diferentes mensajes, el estado del botón y sus combinaciones.

El parámetro `objRange` identifica la parte del documento donde se encuentra el cursor del mouse. Los objetos de intervalo siempre seleccionan una etiqueta entera del documento. (Esto puede cambiar en futuras versiones del producto que ofrezcan un mecanismo de posicionamiento más preciso.) Si en la posición actual del cursor no se encuentra una parte del documento que se pueda seleccionar, el objeto de intervalo es `null`.

NOTA: estos eventos del entorno de scripting y del complemento de Altova XMLSpy para entornos IDE son compatibles pero se vuelven obsoletos con este evento:

```
On_AuthenticMouseMove( )           IXMLSpyPlugIn.OnEvent (15, ...)
    // nEventId = 15
On_AuthenticButtonUp( )          IXMLSpyPlugIn.OnEvent (16, ...)    //
nEventId = 16
On_AuthenticButtonDown( )        IXMLSpyPlugIn.OnEvent (17, ...)
    // nEventId = 17
On_AuthenticButtonDoubleClick( ) IXMLSpyPlugIn.OnEvent (24, ...)    //
nEventId = 24
```

Ejemplos

```
' -----
' fragmento de código VB: conectarse a eventos a nivel de objeto
' -----
' acceder a XMLSpy (sin buscar errores)
Dim objSpy As XMLSpyLib.Application
Set objSpy = GetObject("", "XMLSpy.Application")

' esta es la rutina de devolución de llamada
' conectada al evento OnMouseEvent del objeto objView. Si hace clic con el
botón izquierdo del mouse
' mientras pulsa la tecla Ctrl, la selección actual se asignará
' a la etiqueta en la que está el cursor
Private Function objView_OnMouseEvent(ByVal i_nXPos As Long, ByVal i_nYPos As
Long, ByVal i_eMouseEvent As XMLSpyLib.SPYMouseEvent, ByVal i_pRange As
XMLSpyLib.IAuthenticRange) As Boolean
    If (i_eMouseEvent = (XMLSpyLib.spyLeftButtonDownMask Or
XMLSpyLib.spyCtrlKeyDownMask)) Then
        On Error Resume Next
        i_pRange.Select
        objView_OnMouseEvent = True
    Else
        objView_OnMouseEvent = False
    End If
End Function

' use la palabra clave de VBA WithEvents para conectarse al evento a nivel de
' objeto
Dim WithEvents objView As XMLSpyLib.AuthenticView
Set objView = objSpy.ActiveDocument.AuthenticView
```

```
' continuar con más código ...
' y servir el bucle de mensajes
```

OnSelectionChanged

Evento: OnSelectionChanged (*objNewSelection* as AuthenticRange)

Entorno de scripting de XMLSpy: VBScript:

```
Function On_AuthenticSelectionChanged (objNewSelection)
End Function
```

Entorno de scripting de XMLSpy: JScript:

```
function On_AuthenticSelectionChanged (objNewSelection)
{
}
```

Complemento de XMLSpy para entornos IDE:

```
IXMLSpyPlugIn.OnEvent (23, ...) // nEventId = 23
```

Descripción

Este evento se desencadena cada vez que cambia la selección en la interfaz del usuario.

Ejemplos

```
' -----
' fragmento de código VB: conectarse a eventos a nivel de objeto
' -----
' acceder a XMLSpy (sin buscar errores)
Dim objSpy As XMLSpyLib.Application
Set objSpy = GetObject("", "XMLSpy.Application")

' esta es la rutina de devolución de llamada
' conectada al evento OnSelectionChanged del objeto objView
Private Sub objView_OnSelectionChanged (ByVal i_ipNewRange As
XMLSpyLib.IAuthenticRange)
    MsgBox ("new selection: " & i_ipNewRange.Text)
End Sub

' use la palabra clave de VBA WithEvents para conectarse al evento a nivel de
' objeto
Dim WithEvents objView As XMLSpyLib.AuthenticView
Set objView = objSpy.ActiveDocument.AuthenticView

' continuar con más código ...
' y servir el bucle de mensajes
```

OnToolbarButtonClicked

Evento: OnToolbarButtonClicked (identificador de botón)

Descripción: OnToolbarButtonClicked se desencadena cuando el usuario hace clic en un botón de una barra de herramientas. El parámetro identificador del botón ayuda a determinar en qué botón se hizo clic. A continuación puede consultar la lista predefinida de identificadores de botón:

- AuthenticPrint
- AuthenticPrintPreview
- AuthenticUndo
- AuthenticRedo
- AuthenticCut
- AuthenticCopy
- AuthenticPaste
- AuthenticClear
- AuthenticMarkupHide
- AuthenticMarkupLarge
- AuthenticMarkupMixed
- AuthenticMarkupSmall
- AuthenticValidate
- AuthenticChangeWorkingDBXMLCell
- AuthenticSave
- AuthenticSaveAs
- AuthenticReload
- AuthenticTableInsertRow
- AuthenticTableAppendRow
- AuthenticTableDeleteRow
- AuthenticTableInsertCol
- AuthenticTableAppendCol
- AuthenticTableDeleteCol
- AuthenticTableJoinCellRight
- AuthenticTableJoinCellLeft
- AuthenticTableJoinCellAbove
- AuthenticTableJoinCellBelow
- AuthenticTableSplitCellHorizontally
- AuthenticTableSplitCellVertically
- AuthenticTableAlignCellContentTop
- AuthenticTableCenterCellVertically
- AuthenticTableAlignCellContentBottom
- AuthenticTableAlignCellContentLeft
- AuthenticTableCenterCellContent
- AuthenticTableAlignCellContentRight
- AuthenticTableJustifyCellContent
- AuthenticTableInsertTable
- AuthenticTableDeleteTable
- AuthenticTablePropiedades
- AuthenticAppendRow
- AuthenticInsertRow
- AuthenticDuplicateRow
- AuthenticMoveRowUp
- AuthenticMoveRowDown
- AuthenticDeleteRow
- AuthenticDefineEntities

Además el usuario puede añadir identificadores para sus propios botones. Es necesario tener cuidado al añadir identificadores porque no se comprueba si son únicos. Se pueden usar los mismos identificadores para identificar los botones de las llamadas Set/GetToolbarState() de

la API de COM. Si añade código para cada botón, el usuario tiene la posibilidad de redefinir completamente el comportamiento de la barra de herramientas VistaAuthentic, añadiendo métodos propios para la manipulación de tables, etc.

OnToolbarButtonExecuted

Evento: [OnToolbarButtonExecuted](#) (identificador de botón)

Descripción: OnToolbarButtonClicked se desencadena cuando el usuario hace clic en un botón de una barra de herramientas. El parámetro identificador de botones ayuda a determinar en qué botón se hizo clic. [Consulte aquí](#) la lista predefinida de identificadores de botón.

OnToolbarButtonExecuted se desencadena después de que se ejecuta una acción de barra de herramientas. Por ejemplo, es muy útil a la hora de añadir código actualizado, como en el ejemplo siguiente:

```
//el evento se desencadena cuando se ejecuta una acción de un botón de una barra de herramientas
function On_AuthenticToolbarButtonExecuted( varBtnIdentifier )
{
    UpdateOwnToolbarButtonStates();
}
```

En este caso UpdateOwnToolbarButtonStates es una función del usuario definida en las declaraciones globales.

OnUserAddedXMLNode

Evento: [OnUserAddedXMLNode](#) (nodo XML)

Descripción: OnUserAddedXMLNode se desencadenará cuando el usuario añada un nodo XML como acción principal. Esto ocurre cuando el usuario hace clic en:

- hipervínculos de adición automática.
- en los comandos **Insertar...**, **Insertar después de...**, **Insertar antes de...** de los menús contextuales.
- en los botones **Anexar fila** e **Insertar fila** de la barra de herramientas.
- en las acciones **Insertar después de...**, **Insertar antes de...** del ayudante de entrada de un elemento (fuera de StyleVision).

El evento no se desencadena con el comando **Duplicar fila**, ni cuando se añade el nodo desde fuera de la aplicación (por ejemplo, desde la API de COM), ni con el comando **Aplicar** (por ejemplo, con los iconos de estado del texto), ni cuando se realizan operaciones de tabla XML o de BD.

El parámetro del evento es el objeto nodo XML que se añadió dando el usuario la oportunidad de manipular el nodo XML añadido. En el archivo [OnUserAddedXMLNode.sps](#) puede encontrar un ejemplo avanzado de un controlador de evento (en la carpeta [Authentic/Scripting](#) del proyecto Examples de la ventana Proyecto).

2.15.2 AuthenticView.Application

Propiedad: Application as [Authentic](#) (solo lectura)

Descripción

Acceder al objeto de aplicación de XMLSpy.

Errores

- 2000 El objeto AuthenticView ya no es válido.
- 2005 Se indicó una dirección no válida para el parámetro de devolución.

2.15.3 AuthenticView.AsXMLString

Propiedad: [AsXMLString](#) as String

Descripción

Devuelve o configura el contenido del documento como cadena XML. Al asignarse un nuevo valor al contenido no cambia el archivo de esquema ni el archivo SPS. Si el nuevo XMLString no cumple con el archivo de esquema, se devuelve el error 2011.

Errores

- 2000 El objeto AuthenticView ya no es válido.
- 2011 AsXMLString recibe un valor que no es XML válido según el archivo de esquema actual.

2.15.4 AuthenticView.ContextMenu

Propiedad: [ContextMenu](#) as [ContextMenu](#)

Descripción

La propiedad `ContextMenu` permite personalizar el menú contextual. El mejor sitio para hacerlo es el controlador de eventos `OnContextMenuActivated`.

Errores

- 2000 Objeto no válido.
- 2005 Parámetro no válido.

2.15.5 AuthenticView.CreateXMLNode

Método: `CreateXMLNode (nKind as SPYXMLDataKind) as XMLData`

Valor devuelto

El método devuelve el nuevo objeto [XMLData](#).

Descripción

Para crear un objeto nuevo `XMLData` utilice el método `CreateXMLNode()`.

Errores

- 2000 Objeto no válido.
- 2012 No se puede crear el nodo XML.

2.15.6 AuthenticView.DisableAttributeEntryHelper

Método: `DisableAttributeEntryHelper()`

Descripción

`DisableAttributeEntryHelper()` deshabilita el ayudante de entrada Atributos de XMLSpy, de Authentic Desktop y del complemento Authentic Browser.

Errores

2000 Objeto no válido.

2.15.7 AuthenticView.DisableElementEntryHelper

Método: `DisableElementEntryHelper()`

Descripción

`DisableElementEntryHelper()` deshabilita el ayudante de entrada Elementos de XMLSpy, de Authentic Desktop y del complemento Authentic Browser.

Errores

2000 Objeto no válido.

2.15.8 AuthenticView.DisableEntityEntryHelper

Método: [DisableEntityEntryHelper\(\)](#)

Descripción

`DisableEntityEntryHelper()` deshabilita el ayudante de entrada Entidades de XMLSpy, de Authentic Desktop y del complemento Authentic Browser.

Errores

2000 Objeto no válido.

2.15.9 AuthenticView.DocumentBegin

Propiedad: DocumentBegin as [AuthenticRange](#) (solo lectura)

Descripción

Recupera un objeto de intervalo que apunta al principio del documento.

Errores

- 2000 El objeto AuthenticView ya no es válido.
- 2005 Se indicó una dirección no válida para el parámetro de devolución.

2.15.10 AuthenticView.DocumentEnd

Propiedad: DocumentEnd as [AuthenticRange](#) (solo lectura)

Descripción

Recupera un objeto de intervalo que apunta al final del documento.

Errores

- 2000 El objeto AuthenticView ya no es válido.
- 2005 Se indicó una dirección no válida para el parámetro de devolución.

2.15.11 AuthenticView.DoNotPerformStandardAction

Método: `DoNotPerformStandardAction ()`

Descripción

`DoNotPerformStandardAction()` sirve de burbuja Cancelar para macros. Detiene la ejecución una vez finalizada la macro.

Errores

2000 Objeto no válido.

2.15.12 AuthenticView.EvaluateXPath

Método: [EvaluateXPath \(XMLData, string expression\) as string](#)

Valor devuelto

El método devuelve una cadena de texto.

Descripción

EvaluateXPath() ejecuta una expresión XPath con el nodo de contexto XML dado. El resultado se devuelve en forma de cadena de texto. Si se trata de una secuencia, se devuelve una cadena separada por espacios.

Errores

- 2000 Objeto no válido.
- 2005 Parámetro no válido.
- 2008 Error interno.
- 2013 Error de XPath.

2.15.13 AuthenticView.Event

Propiedad: [Event](#) as AuthenticEvent (solo lectura)

Descripción

Esta propiedad ofrece acceso a los parámetros del último evento, al igual que OldAuthenticView.event. Como todos los eventos para el entorno de scripting y clientes externos ahora tienen parámetros, esta propiedad solo se debería usar desde los complementos para entornos IDE.

Errores

- 2000 El objeto AuthenticView ya no es válido.
- 2005 Se indicó una dirección no válida para el parámetro de devolución.

2.15.14 AuthenticView.EventContext

Propiedad: `EventContext` as `EventContext`

Descripción

`EventContext` ofrece acceso al contexto de las macros que están en ejecución. Consulte la descripción de la interfaz `EventContext` para obtener más información.

Errores

2000 Objeto no válido.

2.15.15 AuthenticView.GetToolbarButtonState

Método: `GetToolbarButtonState (ButtonIdentifier as string) as AuthenticToolbarButtonState`

Valor devuelto

El método devuelve `AuthenticToolbarButtonState`

Descripción

`GetToolbarButtonState` consulta el estado de un botón de una barra de herramientas y permite al usuario deshabilitarlo o habilitarlo. El botón se identifica por medio de su identificador ([consulte esta lista](#)). Por ejemplo, puede usarlo para deshabilitar los botones de las barras de herramientas permanentemente. O puede poner `SetToolbarButtonState` en el controlador de eventos `OnSelectionChanged`, porque los botones de las barras de herramientas se actualizan regularmente cuando en el documento cambia la selección.

Los estados de los botones de las barras de herramientas están en las [enumeraciones](#).

El estado predeterminado significa que la habilitación/deshabilitación del botón está regida por `AuthenticView`. Cuando el usuario configura el estado del botón para habilitarlo/deshabilitarlo, el botón sigue teniendo ese estado hasta que el usuario vuelve a cambiarlo.

Errores

- 2000 Objeto no válido.
- 2005 Parámetro no válido.
- 2008 Error interno.
- 2014 Identificador de botón no válido

2.15.16 AuthenticView.Goto

Método: Goto (*eKind* as SPYAuthenticElementKind, *nCount* as Long, *eFrom* as SPYAuthenticDocumentPosition) as [AuthenticRange](#)

Descripción

Recupera un objeto de intervalo que apunta al principio del elemento nº *nCount* de tipo *eKind*. La posición inicial viene definida por el parámetro *eFrom*. Use valores positivos para *nCount* para navegar hasta el final del documento. Por el contrario, para navegar hasta el principio del documento, utilice valores negativos.

Errores

- | | |
|------|---|
| 2000 | El objeto AuthenticView ya no es válido. |
| 2003 | El destino está después del final del documento. |
| 2004 | El destino está antes del principio del documento. |
| 2005 | El tipo de elemento indicado no es válido.
La posición desde donde se debe comenzar no es <i>spyAuthenticDocumentBegin</i> ni <i>spyAuthenticDocumentEnd</i> . |
| | Se indicó una dirección no válida para el parámetro de devolución. |

Ejemplos

```
Dim objAuthenticView
Set objAuthenticView = objPlugin.AuthenticView

On Error Resume Next
Dim objRange
' ir al principio de la primera tabla del documento
Set objRange = objAuthenticView.Goto (spyAuthenticTable, 1,
spyAuthenticDocumentBegin)
If (Err.number = 0) Then
    objRange.Select()
Else
    MsgBox "No se encontró ninguna tabla en el documento"
End If
```

2.15.17 AuthenticView.IsRedoEnabled

Propiedad: `IsRedoEnabled` as Boolean (solo lectura)

Descripción

True si se puede rehacer la acción y `Redo` es posible.

Errores

- 2000 El objeto AuthenticView ya no es válido.
- 2005 Se indicó una dirección no válida para el parámetro de devolución.

2.15.18 AuthenticView.IsUndoEnabled

Propiedad: `IsUndoEnabled` as Boolean (solo lectura)

Descripción

True si se puede deshacer la acción y `Undo` es posible.

Errores

- 2000 El objeto AuthenticView ya no es válido.
- 2005 Se indicó una dirección no válida para el parámetro de devolución.

2.15.19 AuthenticView.MarkupVisibility

Propiedad: [MarkupVisibility](#) as [SPYAuthenticMarkupVisibility](#)

Descripción

Establecer u obtener la visibilidad actual del marcado.

Errores

- 2000 El objeto AuthenticView ya no es válido.
- 2005 Se indicó un valor de enumeración no válido.
Se indicó una dirección no válida para el parámetro de devolución.

2.15.20 AuthenticView.Parent

Propiedad: Parent as [Authentic](#) (solo lectura)

Descripción

Acceder a la aplicación.

Errores

- 2000 El objeto AuthenticView ya no es válido.
- 2005 Se indicó una dirección no válida para el parámetro de devolución.

2.15.21 AuthenticView.Print

Método: Print (*bWithPreview* as Boolean, *bPromptUser* as Boolean)

Descripción

Imprimir el documento que aparece en la vista. Si *bWithPreview* recibe el valor *True*, aparece el cuadro de diálogo de vista previa de impresión. Si *bPromptUser* recibe el valor *True*, aparece el cuadro de diálogo "Imprimir". Si ambos parámetros reciben el valor *False*, el documento se imprime directamente.

Errores

2000 El objeto AuthenticView ya no es válido.

2.15.22 AuthenticView.Redo

Método: `Redo () as Boolean`

Descripción

Rehacer la modificación deshecha con el último comando Deshacer.

Errores

- 2000 El objeto AuthenticView ya no es válido.
- 2005 Se indicó una dirección no válida para el parámetro de devolución.

2.15.23 AuthenticView.Selection

Propiedad: Selection as [AuthenticRange](#)

Descripción

Establecer u obtener la selección de texto actual de la interfaz del usuario.

Errores

- 2000 El objeto AuthenticView ya no es válido.
- 2002 No hay activa ninguna selección de texto.
- 2005 Se indicó una dirección no válida para el parámetro de devolución.

Ejemplos

```
' -----
'           VBScript
' -----  
  
Dim objAuthenticView  
Set objAuthenticView = objPlugin.AuthenticView  
  
' si estamos al final del documento, volver a comenzar desde el principio  
If (objAuthenticView.Selection.AreEqual(objAuthenticView.DocumentEnd)) Then  
    objAuthenticView.Selection = objAuthenticView.DocumentBegin  
Else  
    ' objAuthenticView.Selection =  
    objAuthenticView.Selection.GotoNextCursorPosition()  
    ' o más breve:  
    objAuthenticView.Selection.GotoNextCursorPosition().Select  
End If
```

2.15.24 AuthenticView.SetToolbarButtonState

Método: `SetToolbarButtonState (ButtonIdentifier as string, AuthenticToolbarButtonState state)`

Descripción

`SetToolbarButtonState` consulta el estado de un botón de una barra de herramientas y permite al usuario deshabilitarlo o habilitarlo. El botón se identifica por medio de su identificador ([consulte esta lista](#)). Por ejemplo, puede usarlo para deshabilitar los botones de las barras de herramientas permanentemente. O puede poner `SetToolbarButtonState` en el controlador de eventos `OnSelectionChanged`, porque los botones de las barras de herramientas se actualizan regularmente cuando en el documento cambia la selección.

Los estados de los botones de las barras de herramientas están en las [enumeraciones](#).

El estado predeterminado significa que la habilitación/deshabilitación del botón está regida por `AuthenticView`. Cuando el usuario configura el estado del botón para habilitarlo/deshabilitarlo, el botón sigue teniendo ese estado hasta que el usuario vuelve a cambiarlo.

Errores

- 2000 Objeto no válido.
- 2008 Error interno.
- 2014 Identificador de botón no válido

2.15.25 AuthenticView.Undo

Método: [Undo \(\) as Boolean](#)

Descripción

Deshacer la última modificación del documento desde dentro de esta vista.

Errores

- 2000 El objeto AuthenticView ya no es válido.
- 2005 Se indicó una dirección no válida para el parámetro de devolución.

2.15.26 AuthenticView.UpdateXMLInstanceEntities

Método: [UpdateXMLInstanceEntities \(\)](#)

Descripción

Actualiza la representación interna de las entidades declaradas y vuelve a llenar el ayudante de entrada. Además, el validador se vuelve a cargar, permitiendo validar correctamente el archivo XML. Recuerde que esto también puede hacer que se vuelvan a cargar los archivos de esquema.

Errores

El método no devuelve nunca un error.

Ejemplo

```
// -----
//          JavaScript
// -----
var objDocType;
objDocType = objPlugin.XMLRoot.GetFirstChild(10);

if(objDocType)
{
    var objEntity = objPlugin.CreateChild(14);
    objEntity.Name = "child";
    objEntity.TextValue = "SYSTEM \"child.xml\" ";
    objDocType.AppendChild(objEntity);

    objPlugin.AuthenticView.UpdateXMLInstanceEntities();
}
```

2.15.27 AuthenticView.WholeDocument

Propiedad: [WholeDocument](#) as [AuthenticRange](#) (solo lectura)

Descripción

Recupera un objeto de intervalo que selecciona todo el documento.

Errores

- 2000 El objeto AuthenticView ya no es válido.
- 2005 Se indicó una dirección no válida para el parámetro de devolución.

2.15.28 AuthenticView.XMLDataRoot

Propiedad: `XMLDataRoot` as `XMLData` (solo lectura)

Descripción

Devuelve o establece el elemento de nivel superior XMLData del documento actual. Este elemento por lo general describe la estructura del documento y sería de tipo `spyXMLDataXMLDocStruct`, `spyXMLDataXMLEntityDocStruct` o `spyXMLDataDTDDocStruct`.

Errores

- 2000 El objeto `AuthenticView` ya no es válido.
- 2005 Se indicó una dirección no válida para el parámetro de devolución.

2.16 AuthenticXMLTableCommands

Métodos

[Insert](#)

[Delete](#)

[AppendRow](#)

[InsertRow](#)

[DeleteRow](#)

[AppendColumn](#)

[InsertColumn](#)

[DeleteColumn](#)

[JoinLeft](#)

[JoinRight](#)

[JoinUp](#)

[JoinDown](#)

[SplitHorizontal](#)

[SplitVertical](#)

[AlignVerticalTop](#)

[AlignVerticalCenter](#)

[AlignVerticalBottom](#)

[AlignHorizontalLeft](#)

[AlignHorizontalRight](#)

[AlignHorizontalCenter](#)

[AlignHorizontalJustify](#)

[EditPropiedades](#)

Propiedades

[MayInsert](#)

[MayDelete](#)

[MayAppendRow](#)

[MayInsertRow](#)

[MayDeleteRow](#)

[MayAppendCol](#)

[MayInsertCol](#)

[MayDeleteCol](#)

[MayJoinLeft](#)

[MayJoinRight](#)

[MayJoinUp](#)

[MayJoinDown](#)

[MaySplitHorizontal](#)

[MaySplitVertical](#)
[MayAlignVertical](#)
[MayAlignHorizontal](#)
[MayEditPropiedades](#)

2.16.1 AuthenticXMLTableCommands.AlignHorizontalCenter

Declaración AlignHorizontalCenter()

Descripción

Asigna el valor `center` al atributo de alineación horizontal o borra el atributo si ya tenía asignado el valor `center`.

2.16.2 AuthenticXMLTableCommands.AlignHorizontalJustify

Declaración AlignHorizontalJustify()

Descripción

Asigna el valor `justify` al atributo de alineación horizontal o borra el atributo si ya tenía asignado el valor `justify`.

2.16.3 AuthenticXMLTableCommands.AlignHorizontalLeft

Declaración AlignHorizontalLeft()

Descripción

Asigna el valor `left` al atributo de alineación horizontal o borra el atributo si ya tenía asignado el valor `left`.

2.16.4 AuthenticXMLTableCommands.AlignHorizontalRight

Declaración AlignHorizontalRight()

Descripción

Asigna el valor `right` al atributo de alineación horizontal o borra el atributo si ya tenía asignado el valor `right`.

2.16.5 AuthenticXMLTableCommands.AlignVerticalBottom

Declaración AlignVerticalBottom()

Descripción

Asigna el valor `bottom` al atributo de alineación vertical o borra el atributo si ya tenía asignado el valor `bottom`.

2.16.6 AuthenticXMLTableCommands.AlignVerticalCenter

Declaración AlignVerticalCenter()

Descripción

Asigna el valor `center` al atributo de alineación vertical o borra el atributo si ya tenía asignado el valor `center`.

2.16.7 AuthenticXMLTableCommands.AlignVerticalTop

Declaración AlignVerticalTop()

Descripción

Asigna el valor `top` al atributo de alineación vertical o borra el atributo si ya tenía asignado el valor `top`.

2.16.8 AuthenticXMLTableCommands.AppendCol

Declaración AppendCol()

Descripción

Anexa una columna a la tabla actual.

2.16.9 AuthenticXMLTableCommands.AppendRow

Declaración [AppendRow\(\)](#)

Descripción

Anexa una fila a la tabla actual.

2.16.10 AuthenticXMLTableCommands.Delete

Declaración [Delete\(\)](#)

Descripción

Si el usuario confirmó el cuadro de diálogo, el método elimina la tabla que está seleccionada.

2.16.11 AuthenticXMLTableCommands.DeleteCol

Declaración DeleteCol()

Descripción

El método elimina la columna que está seleccionada.

Si solamente queda una columna y el usuario confirma el cuadro de diálogo, se elimina la tabla entera.

2.16.12 AuthenticXMLTableCommands.DeleteRow

Declaración [DeleteRow\(\)](#)

Descripción

El método elimina la fila que está seleccionada.

Si solamente queda una fila y el usuario confirma el cuadro de diálogo, se elimina la tabla entera.

2.16.13 AuthenticXMLTableCommands.EditProperties

Declaración [EditPropiedades\(\)](#)

Descripción

Muestra el cuadro de diálogo "Propiedades" de la celda/tabla seleccionada.

2.16.14 AuthenticXMLTableCommands.Insert

Declaración [Insert\(\)](#)

Descripción

Muestra un cuadro de diálogo e inserta una tabla nueva en el XML actual.

2.16.15 AuthenticXMLTableCommands.InsertCol

Declaración [InsertCol\(\)](#)

Descripción

Inserta una columna nueva antes de la columna que está seleccionada.

2.16.16 AuthenticXMLTableCommands.InsertRow

Declaración [InsertRow\(\)](#)

Descripción

Inserta una fila nueva antes de la fila que está seleccionada.

2.16.17 AuthenticXMLTableCommands.JoinDown

Declaración [JoinDown\(\)](#)

Descripción

Combina la celda actual con la celda situada justo debajo.

2.16.18 AuthenticXMLTableCommands.JoinLeft

Declaración [JoinLeft\(\)](#)

Descripción

Combina la celda actual con la celda situada a su izquierda.

2.16.19 AuthenticXMLTableCommands.JoinRight

Declaración [JoinRight\(\)](#)

Descripción

Combina la celda actual con la celda situada a su derecha.

2.16.20 AuthenticXMLTableCommands.JoinUp

Declaración [JoinUp\(\)](#)

Descripción

Combina la celda actual con la celda situada justo encima.

2.16.21 AuthenticXMLTableCommands.MayAlignHorizontal

Declaración MayAlignHorizontal as Boolean

Descripción

True si se puede establecer el valor de los atributos de alineación horizontal para la celda actual.

2.16.22 AuthenticXMLTableCommands.MayAlignVertical

Declaración MayAlignVertical as Boolean

Descripción

True si se puede establecer el valor de los atributos de alineación vertical para la celda actual.

2.16.23 AuthenticXMLTableCommands.MayAppendCol

Declaración MayAppendCol as Boolean

Descripción

True si se puede anexar una columna.

2.16.24 AuthenticXMLTableCommands.MayAppendRow

Declaración MayAppendRow as Boolean

Descripción

True si se puede anexar una fila.

2.16.25 AuthenticXMLTableCommands.MayDelete

Declaración MayDelete as Boolean

Descripción

True si hay seleccionada una tabla.

2.16.26 AuthenticXMLTableCommands.MayDeleteCol

Declaración MayDeleteCol as Boolean

Descripción

True si la columna actual se puede eliminar.

2.16.27 AuthenticXMLTableCommands.MayDeleteRow

Declaración MayDeleteRow as Boolean

Descripción

True si la fila actual se puede eliminar.

2.16.28 AuthenticXMLTableCommands.MayEditProperties

Declaración MayEditPropiedades as Boolean

Descripción

True si el cuadro de diálogo "Propiedades" está disponible para la celda/tabla seleccionada.

2.16.29 AuthenticXMLTableCommands.MayInsert

Declaración MayInsert as Boolean

Descripción

True si se puede insertar una tabla nueva en la selección actual.

2.16.30 AuthenticXMLTableCommands.MayInsertCol

Declaración MayInsertCol as Boolean

Descripción

True si se puede insertar una columna.

2.16.31 AuthenticXMLTableCommands.MayInsertRow

Declaración MayInsertRow as Boolean

Descripción

True si se puede insertar una fila.

2.16.32 AuthenticXMLTableCommands.MayJoinDown

Declaración MayJoinDown as Boolean

Descripción

True si la celda actual se puede combinar con la celda situada justo debajo.

2.16.33 AuthenticXMLTableCommands.MayJoinLeft

Declaración MayJoinLeft as Boolean

Descripción

True si la celda actual se puede combinar con la celda situada a la izquierda.

2.16.34 AuthenticXMLTableCommands.MayJoinRight

Declaración MayJoinRight as Boolean

Descripción

True si la celda actual se puede combinar con la celda situada a la derecha.

2.16.35 AuthenticXMLTableCommands.MayJoinUp

Declaración MayJoinUp as Boolean

Descripción

True si la celda actual se puede combinar con la celda situada justo encima.

2.16.36 AuthenticXMLTableCommands.MaySplitHorizontal

Declaración MaySplitHorizontal as Boolean

Descripción

True si la celda actual se puede dividir horizontalmente.

2.16.37 AuthenticXMLTableCommands.MaySplitVertical

Declaración MaySplitVertical as Boolean

Descripción

True si la celda actual se puede dividir verticalmente.

2.16.38 AuthenticXMLTableCommands.SplitHorizontal

Declaración [SplitHorizontal\(\)](#)

Descripción

El método divide la celda actual horizontalmente.

2.16.39 AuthenticXMLTableCommands.SplitVertical

Declaración [SplitVertical\(\)](#)

Descripción

El método divide la celda actual verticalmente.

2.17 XMLData

Métodos

[InsertChild](#)

[AppendChild](#)

[EraseAllChildren](#)

[EraseCurrentChild](#)

[GetCurrentChild](#)

[GetFirstChild](#)

[GetNextChild](#)

[GetChild](#)

[GetChildKind](#)

[IsSameNode](#)

[HasChildrenKind](#)

[CountChildren](#)

[CountChildrenKind](#)

Propiedades

[Name](#)

[TextValue](#)

[HasChildren](#)

[MayHaveChildren](#)

[Kind](#)

[Parent](#)

Descripción

Puede usar la interfaz XMLData para manipular el contenido del XML que está visible. Esta interfaz es un equivalente COM más sencillo de la implementación utilizada dentro del complemento y de Altova XMLSpy.

Para crear un objeto XMLData nuevo utilice el método CreateChild() de la interfaz del complemento.

2.17.1 XMLData.AppendChild

Declaración AppendChild(*pNewData* as XMLData)

Descripción

AppendChild anexa pNewData como último elemento secundario en el objeto XMLData. Consulte también [Usar XMLData](#).

Ejemplo

```
Dim objCurrentParent  
Dim objNewChild  
  
Set objNewChild = objPlugIn.CreateChild(spyXMLElement)  
Set objCurrentParent = objPlugIn.XMLRoot  
  
objCurrentParent.AppendChild objNewChild  
  
Set objNewChild = Nothing
```

2.17.2 XMLData.CountChildren

Declaración CountChildren as long

Descripción

CountChildren obtiene el número de elementos secundarios.

Disponible con TypeLibrary versión 1.6

Errores

1500 El objeto XMLData ya no es válido.

2.17.3 XMLData.CountChildrenKind

Declaración CountChildrenKind (*nKind* as [SPYXMLDataKind](#)) as long

Descripción

CountChildrenKind obtiene el número de secundarios del tipo indicado.

Disponible con TypeLibrary versión 1.6

Errores

1500 El objeto XMLData ya no es válido.

2.17.4 XMLData.EraseAllChildren

Declaración [EraseAllChildren](#)

Descripción

EraseAllChildren elimina todos los secundarios asociados del objeto XMLData.

Ejemplo

En el ejemplo se eliminan todos los elementos del documento activo.

```
Dim objCurrentParent  
  
Set objCurrentParent = objPlugIn.XMLRoot  
objCurrentParent.EraseAllChildren
```

2.17.5 XMLData.EraseChild

Método: [EraseChild](#) (Nodo secundario y nodo nuevo como [XMLData](#))

Descripción

Elimina el nodo secundario dado.

Errores

- 1500 Objeto no válido.
- 1506 XML de entrada no válido
- 1510 Parámetro no válido.

2.17.6 XMLData.EraseCurrentChild

Declaración [EraseCurrentChild](#)

Descripción

EraseCurrentChild elimina el objeto secundario XMLData actual. Antes de llamar a EraseCurrentChild debe inicializar un iterador interno con [XMLData.GetFirstChild](#).

Ejemplo

Este ejemplo de JavaScript elimina todos los elementos llamados "Bórrame". En el código puede ver que se puede llamar a EraseCurrentChild y a GetNextChild dentro del mismo bucle para seguir recorriendo los elementos secundarios.

```
function DeleteXMLElements(objXMLData)
{
    if(objXMLData == null)
        return;

    if(objXMLData.HasChildren) {
        var objChild;
        objChild = objXMLData.GetFirstChild(-1);

        while(objChild) {
            DeleteXMLElements(objChild);

            try {
                if(objChild.Name == "Bórrame")
                    objXMLData.EraseCurrentChild();

                objChild = objXMLData.GetNextChild();
            }
            catch(Err) {
                objChild = null;
            }
        }
    }
}
```

2.17.7 XMLData.GetChild

Declaración `GetChild (position as long) as XMLData`

Valor devuelto

Devuelve un elemento XML como objeto XMLData.

Descripción

`GetChild()` devuelve una referencia al secundario en el índice dado (basado en cero).

Disponible con TypeLibrary versión 1.6

Errores

- 1500 El objeto XMLData ya no es válido.
- 1510 Se indicó una dirección no válida para el parámetro de devolución.

2.17.8 XMLData.GetChildAttribute

Método: [GetChildAttribute](#) (Nombre como cadena) como objeto XMLData (NULL cuando haya error)

Descripción

Recupera el atributo que tiene el nombre dado.

Errores

- 1500 Objeto no válido.
- 1510 Parámetro no válido.

2.17.9 XMLData.GetChildElement

Método: [GetChildElement](#) (Nombre como cadena, posición como entero) como objeto XMLData (NULL cuando haya error)

Descripción

Recupera el elemento secundario n que tiene el nombre dado.

Errores

- 1500 Objeto no válido.
- 1510 Parámetro no válido.

2.17.10 XMLData.GetChildKind

Declaración `GetChildKind (position as long, nKind as SPYXMLDataKind) as XMLData`

Valor devuelto

Devuelve un elemento XML como objeto XMLData.

Descripción

`GetChildKind()` devuelve una referencia a un secundario de su tipo en el índice dado (basado en cero). El parámetro de posición es relativo al número de secundarios del tipo indicado y no a todos los secundarios del objeto.

Disponible con TypeLibrary versión 1.6

Errores

- 1500 El objeto XMLData ya no es válido.
- 1510 Se indicó una dirección no válida para el parámetro de devolución.

2.17.11 XMLData.GetCurrentChild

Declaración `GetCurrentChild` as [XMLData](#)

Valor devuelto

Devuelve un elemento XML como objeto XMLData.

Descripción

`GetCurrentChild` obtiene el secundario actual. Antes de llamar a `GetCurrentChild` debe inicializar un iterador interno con [XMLData.GetFirstChild](#).

2.17.12 XMLData.GetChildFirst

Declaración GetFirstChild(*nKind* as [SPYXMLDataKind](#)) as [XMLData](#)

Valor devuelto

Devuelve un elemento XML como objeto XMLData.

Descripción

GetFirstChild inicializa un iterador nuevo y devuelve el primer secundario. Asigne el valor nKind = -1 para obtener un iterador para todos los tipos de secundarios.

Ejemplo

Consulte el ejemplo disponible en [XMLData.GetNextChild](#).

2.17.13 XMLData.GetNamespacePrefixForURI

Método: [GetNamespacePrefixForURI](#) (URI como cadena de texto) Prefijo como cadena de texto

Descripción

Devuelve el prefijo de espacio de nombres del URI dado.

Errores

- 1500 Objeto no válido.
- 1510 Parámetro no válido.

2.17.14 XMLData.GetNextChild

Declaración GetNextChild as [XMLData](#)

Valor devuelto

Devuelve un elemento XML como objeto XMLData.

Descripción

GetNextChild pasa al siguiente secundario de este elemento. Antes de llamar a GetNextChild debe inicializar un iterador interno con [XMLData.GetFirstChild](#).

Busque el último secundario del elemento tal y como muestra el ejemplo siguiente.

Ejemplo

```
On Error Resume Next
Set objParent = objPlugIn.XMLRoot

'obtener elementos de todos los tipos
Set objCurrentChild = objParent.GetFirstChild(-1)

Do
    'hacer algo útil con el secundario

    'pasar al siguiente secundario
    Set objCurrentChild = objParent.GetNextChild
Loop Until (Err.Number - vbObjectError = 1503)
```

2.17.15 XMLData.GetTextValueXMLDecoded

Método: `GetTextValueXMLDecoded` as string

Descripción

Obtiene el valor de texto codificado del XML.

Errores

- 1500 Objeto no válido.
- 1510 Parámetro no válido.

2.17.16 XMLData.HasChildren

Declaración HasChildren as Boolean

Descripción

La propiedad es true si el objeto es primario de otros objetos XMLData.

Esta propiedad es de solo lectura.

2.17.17 XMLData.HasChildrenKind

Declaración HasChildrenKind (*nKind* as [SPYXMLDataKind](#)) as Boolean

Descripción

El método devuelve true si el objeto es el primario de otros objetos XMLData del mismo tipo.

Disponible con TypeLibrary versión 1.6

Errores

- 1500 El objeto XMLData ya no es válido.
- 1510 Se indicó una dirección no válida para el parámetro de devolución.

2.17.18 XMLData.InsertChild

Declaración `InsertChild(pNewData as XMLData)`

Descripción

`InsertChild` inserta el secundario nuevo antes del secundario actual.

Consulte también [XMLData.GetFirstChild](#), [XMLData.GetNextChild](#) para establecer el secundario actual.

2.17.19 XMLData.InsertChildAfter

Método: [InsertChildAfter](#) (Nodo secundario y nodo nuevo como XMLData)

Descripción

Inserta un nodo XML nuevo después del nodo dado.

Errores

- 1500 Objeto no válido.
- 1506 XML de entrada no válido
- 1507 No se admiten elementos secundarios
- 1510 Parámetro no válido.
- 1512 El secundario ya se añadió
- 1514 Tipo no válido en la posición

2.17.20 XMLData.InsertChildBefore

Método: [InsertChildBefore](#) (Nodo secundario y nodo nuevo como XMLData)

Descripción

Inserta un nodo XML nuevo antes del nodo dado.

Errores

- 1500 Objeto no válido.
- 1506 XML de entrada no válido
- 1507 No se admiten elementos secundarios
- 1510 Parámetro no válido.
- 1512 El secundario ya se añadió
- 1514 Tipo no válido en la posición

2.17.21 XMLData.IsSameNode

Declaración `IsSameNode(pNodeToCompare as XMLData) as Boolean`

Descripción

Devuelve true si pNodeToCompare hace referencia al mismo nodo que el objeto mismo.

2.17.22 XMLData.Kind

Declaración Kind as [SPYXMLDataKind](#)

Descripción

El tipo de este objeto XMLData.

Esta propiedad es de solo lectura.

2.17.23 XMLData.MayHaveChildren

Declaración MayHaveChildren as Boolean

Descripción

Comprueba si se pueden añadir secundarios a este objeto XMLData.

Esta propiedad es de solo lectura.

2.17.24 XMLData.Name

Declaración Name as String

Descripción

Se utiliza para modificar y obtener el nombre del objeto XMLData.

2.17.25 XMLData.Parent

Declaración Parent as [XMLData](#)

Valor devuelto

El primario como como objeto XMLData.

Nada (o NULL) si no hay elemento primario.

Descripción

Elemento primario de este elemento.

Esta propiedad es de solo lectura.

2.17.26 XMLData.SetTextValueXMLDecoded

Método: [SetTextValueXMLDecoded](#) (valor de cadena)

Descripción

Establece el valor de texto codificado del XML.

Errores

- 1500 Objeto no válido.
- 1513 La modificación no está permitida.

2.17.27 XMLData.TextValue

Declaración `TextValue as String`

Descripción

Se utiliza para modificar y para obtener el valor de texto de este objeto XMLData. Consulte también [Usar XMLData](#).

3 Enumeraciones

Esta sección enumera y describe las enumeraciones de Authentic Browser.

3.1 SPYAuthenticActions

Descripción

Acciones que se pueden llevar a cabo en objetos [AuthenticRange](#).

Valores posibles:

spyAuthenticInsertAt	= 0
spyAuthenticApply	= 1
spyAuthenticClearSurr	= 2
spyAuthenticAppend	= 3
spyAuthenticInsertBefore	= 4
spyAuthenticRemove	= 5

3.2 SPYAuthenticCommand

Descripción

Enumeración de todos los comandos disponibles

Valores posibles:

```
// CommandGroupMain
    k_CommandSeparator      = 0
    k_CommandSave           = 1
    k_CommandPrint          = 2
    k_CommandPrintPreview   = 3
    k_CommandValidate       = 4
    k_CommandUndo           = 5
    k_CommandRedo           = 6

// CommandGroupEdit
    k_CommandEditCut        = 7
    k_CommandEditCopy       = 8
    k_CommandEditPaste      = 9
    k_CommandEditFind       = 10
    k_CommandEditRepeat     = 11
    k_CommandEditReplace    = 12

// CommandGroupMarkup
    k_CommandMarkupHide     = 13
    k_CommandMarkupLarge    = 14

// CommandGroupRow
    k_CommandRowAppend      = 15
    k_CommandRowInsert       = 16
    k_CommandRowDuplicate   = 17
    k_CommandRowMoveUp      = 18
    k_CommandRowMoveDown    = 19
    k_CommandRowDelete      = 20

// CommandGroupXMLTables
    k_CommandXMLTableInsert = 21
    k_CommandXMLTableDelete = 22
    e
    k_CommandXMLTableAppe   = 23
    ndRow
    k_CommandXMLTableInsert = 24
    Row
    k_CommandXMLTableDelete = 25
    eRow
    k_CommandXMLTableAppe   = 26
    ndCol
    k_CommandXMLTableInsert = 27
    Col
    k_CommandXMLTableDelete = 28
    eCol
    k_CommandXMLTableJoinRi= 29
```

```
ght
k_CommandXMLTableJoinL = 30
eft
k_CommandXMLTableJoinU = 31
p
k_CommandXMLTableJoinD = 32
own
k_CommandXMLTableSplitH= 33
orz
k_CommandXMLTableSplitV= 34
ert
k_CommandXMLTableVAlig = 35
nTop
k_CommandXMLTableVAlig = 36
nCenter
k_CommandXMLTableVAlig = 37
nBottom
k_CommandXMLTableAlign = 38
Left
k_CommandXMLTableAlign = 39
Center
k_CommandXMLTableAlign = 40
Right
k_CommandXMLTableAlignJ= 41
ustify
k_CommandXMLTableEditP = 42
ropiedades

// since TypeLib 1.2
k_CommandCheckSpelling      = 43
k_CommandAbout               = 44
k_CommandPackageManagement   = 45
```

3.3 SPYAuthenticCommandGroup

Descripción

Grupos a los que puede pertenecer un comando.

Valores posibles:

k_CommandGroupMain	= 0
k_CommandGroupEdit	= 1
k_CommandGroupMarkup	= 2
k_CommandGroupRow	= 3
k_CommandGroupXMLTables	= 4

3.4 SPYAuthenticDocumentPosition

Descripción

Posiciones relativas y absolutas utilizadas para navegar con objetos [AuthenticRange](#).

Valores posibles:

spyAuthenticDocumentBegin	= 0
spyAuthenticDocumentEnd	= 1
spyAuthenticRangeBegin	= 2
spyAuthenticRangeEnd	= 3

3.5 SPYAuthenticElementActions

Descripción

Acciones que se pueden usar con [GetAllowedElements](#).

Valores posibles:

k_ActionInsertAt	= 0
k_ActionApply	= 1
k_ActionClearSurr	= 2
k_ActionAppend	= 3
k_ActionInsertBefore	= 4
k_ActionRemove	= 5

3.6 SPYAuthenticElementKind

Descripción

Enumeración de los diferentes tipos de elemento utilizados para navegación y selección dentro de objetos [AuthenticRange](#) y [AuthenticView](#).

Valores posibles:

spyAuthenticChar	= 0
spyAuthenticWord	= 1
spyAuthenticLine	= 3
spyAuthenticParagraph	= 4
spyAuthenticTag	= 6
spyAuthenticDocument	= 8
spyAuthenticTable	= 9
spyAuthenticTableRow	= 10
spyAuthenticTableColumn	= 11

3.7 SPYAuthenticEntryHelperWindows

Descripción

Identificación de las ventanas de ayudantes de entrada disponibles.

Valores posibles:

k_Elements	= 1
k_Attributes	= 2
k_Entities	= 4

3.8 SPYAuthenticMarkupVisibility

Descripción

Valores de enumeración para personalizar la visibilidad del marcado.

Valores posibles:

spyAuthenticMarkupHidden	= 0
spyAuthenticMarkupSmall	= 1
spyAuthenticMarkupLarge	= 2
spyAuthenticMarkupMixed	= 3

3.9 SPYAuthenticToolbarAlignment

Descripción

Valores para especificar la alineación de la barra de herramientas.

Valores posibles:

k_ToolbarAlignTop	= 0
k_ToolbarAlignLeft	= 1
k_ToolbarAlignBottom	= 2
k_ToolbarAlignRight	= 3

3.10 SPYAuthenticToolbarButtonState

Descripción

Estados de los botones de la barra de herramientas de Authentic:

Valores posibles:

authenticToolbarButtonDefault	= 0
authenticToolbarButtonEnabled	= 1
authenticToolbarButtonDisabled	= 2

3.11 SPYXMLDataKind

Descripción

Los diferentes tipos de elementos XMLData disponibles para documentos XML.

Valores posibles:

spyXMLDataXMLDocStruct	= 0
spyXMLDataXMLEntityDocStruct	= 1
spyXMLDataDTDDocStruct	= 2
spyXMLDataXML	= 3
spyXMLDataElement	= 4
spyXMLDataAttr	= 5
spyXMLDataText	= 6
spyXMLDataCData	= 7
spyXMLDataComment	= 8
spyXMLDataP	= 9
spyXMLDataDefDoctype	= 10
spyXMLDataDefExternalID	= 11
spyXMLDataDefElement	= 12
spyXMLDataDefAttlist	= 13
spyXMLDataDefEntity	= 14
spyXMLDataDefNotation	= 15
spyXMLDataKindsCount	= 16

Altova Authentic 2015 Browser Edition

Aplicaciones web ASP.NET

Aplicaciones web ASP.NET

A fin de facilitar la implementación de **Altova® Authentic® Browser Edition** ofrecemos un control de servidor ASP.NET que integra Visual Studio .NET completamente y permite a los desarrolladores de ASP .NET arrastrar y colocar **Authentic Browser** en sus formularios web.

Para usar el control de servidor ASP .NET

1. Descargue el [**control de servidor ASP.NET para el complemento Authentic Browser**](#) del sitio web de Altova.
2. Inicie Visual Studio y registre el ensamblado (el paquete descargado) en Visual Studio .NET añadiéndolo al Cuadro de herramientas. Esto se hace seleccionando **Herramientas | Elegir elementos del cuadro de herramientas**, seleccionando *Componentes de .NET Framework* en el cuadro de diálogo "Elementos del cuadro de herramientas" y navegando hasta el ensamblado (Altova\Authentic.WebControls.dll).
3. El control **AuthenticDocumentView** aparece en el panel *General* del Cuadro de herramientas. Arrastre este control hasta la vista de diseño y ajuste su tamaño o modifíquelo según corresponda. (El control invocará a **Authentic Browser** en la página web del lado cliente.)
4. Añada al proyecto los archivos relacionados con el complemento (archivos de esquema, archivo SPS, archivos XML, archivos de imagen, etc.). Esto se puede hacer añadiendo una carpeta nueva al proyecto de Visual Studio desde el explorador de soluciones y añadiendo los archivos relacionados con el complemento a la nueva carpeta de proyecto. Para cada uno de estos archivos configure estas propiedades: **Acción de generación = Contenido** y **Copiar en el directorio de resultados = Sí, cuando sea más reciente**.
5. Configure estas propiedades para el control **AuthenticDocumentView**: (i) **Seleccionar la versión Trusted/Untrusted**; (ii) Agregar los recursos relacionados con el SPS (*SchemaDataURL*, *SPSDataURL*, *XMLDataURL*); (iii) si se trata de la edición Enterprise Edition, añada la información sobre la licencia.
6. Para IIS 7.0 o superior, añada un controlador de contenido estático para el archivo SPS. Esto se hace añadiendo estas líneas de código al archivo *web.config*:

```
<system.webServer>
  <staticContent>
    <mimeType fileExtension=".sps" mimeType="text/xml" />
  </staticContent>
</system.webServer>
```

Ejemplo de proyecto ASP.NET

En el paquete del control de servidor ASP .NET hay un ejemplo de proyecto de aplicación web ASP .NET para Visual Studio 2008 llamado **QuickStartAppl**. Se encuentra en la carpeta **Example** del paquete descomprimido.

Altova Authentic 2015 Browser Edition

Información sobre licencias

Información sobre licencias

Esta sección incluye:

- información sobre la [distribución de este producto de software](#).
- información sobre los [derechos de propiedad intelectual](#) relacionados con este producto de software.
- el [contrato de licencia para el usuario final de Authentic](#) que rige el uso de este producto de software

Los términos del contrato de licencia que aceptó al instalar el producto de software son vinculantes, por lo que rogamos lea atentamente toda esta información.

1 Distribución electrónica de software

Este producto está disponible por distribución electrónica de software, un método de distribución que ofrece ventajas únicas:

- Puede evaluar el software de forma totalmente gratuita antes de decidir si compra el producto.
- Si decide comprarlo, puede hacer un pedido en línea en el [sitio web de Altova](#) y conseguir en pocos minutos el software con licencia.
- Si realiza el pedido en línea, siempre recibirá la versión más reciente de nuestro software.
- El paquete de instalación del producto incluye un sistema de ayuda en pantalla totalmente integrado. La versión más reciente del manual del usuario está disponible en www.altova.com (i) en formato HTML y (ii) en formato PDF para descargar e imprimir si lo desea.

Período de evaluación de 30 días

Después de descargar el producto de software, puede probarlo de forma totalmente gratuita durante un plazo de 30 días. Pasados unos 20 días, el software empieza a recordarle que no tiene una licencia. El mensaje de aviso aparece una sola vez al iniciarse la aplicación. Si desea usar el programa una vez pasado el plazo de 30 días, deberá comprar una licencia permanente y aceptar el [contrato de licencia de software de Altova](#), que se entrega en forma de código clave. La licencia puede comprarse directamente en la tienda en línea del [sitio web de Altova](#). Después de comprar la licencia recibirá el código clave, que debe introducir en el cuadro de diálogo "Activación del software" para desbloquear el producto de forma permanente.

Distribuir la versión de evaluación a otros usuarios de su organización

Si desea distribuir la versión de evaluación en la red de su compañía o si desea usarlo en un PC que no está conectado a Internet, solamente puede distribuir los programas de instalación (siempre y cuando no se modifiquen de forma alguna). Todo usuario que acceda al instalador debe solicitar su propio código clave de evaluación (de 30 días). Una vez pasado este plazo de 30 días, todos los usuarios deben comprar también una licencia para poder seguir usando el producto.

Para más información consulte el [contrato de licencia de software de Altova](#) que aparece al final de esta sección.

2 Derechos de propiedad intelectual

El software de Altova y sus copias (si tiene permiso de Altova para realizar copias) es propiedad intelectual de Altova y de sus proveedores. La estructura, la organización y el código del software se considera secreto comercial e información confidencial de Altova y de sus proveedores. El software está protegido por las leyes de derechos de autor, como la ley de derechos de autor de EE UU, tratados internacionales y la legislación vigente del país donde se utiliza, entre otras. Altova conserva los derechos de propiedad de todas las patentes, derechos de autor, secretos comerciales, marcas registradas y otros derechos de propiedad intelectual pertenecientes al software y los derechos de propiedad de Altova abarcan también imágenes, fotografías, animaciones, videos, audio, música, texto y otros applets incorporados al software y al material impreso que viene con el software. Las notificaciones de infracción de dichos derechos de autor debe enviarse al agente de derechos de autor de Altova, cuyos datos de contacto aparecen en el sitio web de Altova.

El software de Altova contiene software de terceros que también está protegido por las leyes de propiedad intelectual, incluida, entre otras, la legislación de derechos de autor mencionada en http://www.altova.com/es/legal_3rdparty.html.

Los demás nombres y marcas registradas son propiedad de sus respectivos propietarios.

3 Altova Authentic: contrato de licencia para el usuario final

Altova End-User License Agreement for Authentic
THIS IS A LEGAL DOCUMENT -- RETAIN FOR YOUR RECORDS
ALTOVA® END-USER LICENSE AGREEMENT
For Authentic® Enterprise Software Editions
And Authentic® Community Software Editions

Licensor:
Altova GmbH
Rudolfsplatz 13a/9
A-1010 Wien
Austria

Important - Read Carefully. Notice to User:

This Altova End User License Agreement for Authentic® (“AEULA”) governs your right to (i) use the Authentic Desktop Enterprise Edition software, (ii) use, reproduce and distribute the Authentic Browser-Plugin Enterprise Edition software, (iii) use the Authentic Desktop Community Edition software and (iv) use, reproduce and distribute the Authentic Browser-Plugin Community Edition software (each or collectively hereinafter referenced, as “Authentic Software”). Your license rights depend on the specific software edition that you have licensed as some editions have different rights and restrictions applicable to them as set forth in detail below. This Authentic EULA is a legal document between you and Altova GmbH (“Altova”). It is important that you read this document before using the Altova-provided software and any accompanying documentation, including, without limitation, printed materials, ‘online’ files, or electronic documentation (“Documentation”). By clicking the “I accept” and “Next” buttons below, or by installing, or otherwise using the Authentic Software, you agree to be bound by the terms of this AEULA as well as the Altova Privacy Policy (“Privacy Policy”) including, without limitation, the warranty disclaimers, limitation of liability, data use and termination provisions below. You agree that this agreement is enforceable like any written agreement negotiated and signed by you. If you do not agree, you are not licensed to use the Authentic Software, and you must destroy any downloaded copies of the Authentic Software in your possession or control. Please go to our Web site at <http://www.altova.com/authenticeula> to download and print a copy of this Authentic EULA for your files and <http://www.altova.com/privacy> to review the Privacy Policy.

1. Authentic Desktop Enterprise Edition (“ADEE”) Software Terms and Conditions

(a) **License Grant.** Upon your acceptance of this AEULA, Altova grants you a non-exclusive, non-transferable (except as provided below), limited license, without the right to grant sublicenses, to install and use a copy of ADEE software on one compatible personal computer or workstation up to the Permitted Number of computers. You may not distribute or reproduce the ADEE software other than as expressly permitted. Subject to the limitations set forth in Section 1(a)(i) you may install and use a copy of this software on more than one of your compatible personal computers or workstations if you have purchased a Named-User license. Subject to the limitations set forth in Sections 1(a)(iii) and 1(a)(iv), users may use the software concurrently on a network. The Permitted Number of computers and/or users and the type of license, e.g. Installed, Named-User, and Concurrent-User, shall be determined and specified at such time as you elect to purchase the software. Installed user licenses are intended to be fixed and not concurrent. In other words, you cannot uninstall for one user in order to reinstall that license to a different user

and then uninstall and reinstall back to the original user. Users should be static. Notwithstanding the foregoing, permanent switchovers are acceptable (i.e., an employee has left the company, machine is retired). During the evaluation period, hereinafter defined, only a single user may install and use the ADEE software on one (1) personal computer or workstation. You may install one (1) copy of the ADEE software on a computer file server within your internal network solely for the purpose of downloading and installing this ADEE software onto other computers within your internal network up to the Permitted Number of computers in a commercial environment only. No other network use is permitted, including without limitation using the ADEE software either directly or through commands, data or instructions from or to a computer not part of your internal network, for Internet or Web-hosting services or by any user not licensed to use this copy of ADEE software through a valid license from Altova, except as set forth in Sections 1(a)(iii) and (iv) below Altova makes no warranties or representations about the performance of Altova software in a terminal server environment and the foregoing are expressly excluded from the limited warranty in Section 3(c) hereof and technical support is not available with respect to issues arising from use in such an environment.

(i) If you have licensed the "Named-User" version of the ADEE software, you may install the software on up to five (5) compatible personal computers or workstations of which you are the primary user thereby allowing you to switch from one computer to the other as necessary provided that only one (1) instance of the ADEE software will be used by you as the Named-User at any given time. If you have purchased multiple Named-User licenses, each individual Named-User will receive a separate license key code.

(ii) If you have licensed a "Concurrent-User" version of the ADEE software, you may install the ADEE software on any compatible computers in a commercial environment only, up to ten (10) times the Permitted Number of users, provided that only the Permitted Number of users actually use the software at the same time and further provided that the computers on which the ADEE software is installed are on the same physical computer network. The Permitted Number of concurrent users shall be delineated at such time as you elect to purchase the ADEE licenses. Each separate physical network or office location requires its own set of separate Concurrent User Licenses for those wishing to use the Concurrent-User versions of the software in more than one location or on more than one network, all subject to the above Permitted Number limitations and based on the number of users using or needing access to the software. If a computer is not on the same physical network, then a locally installed user license or a license dedicated to concurrent use in a virtual environment is required.

(iii) If you have licensed a "Concurrent-User" versions of ADEE software, you may install a copy of the ADEE Software on a terminal server (Microsoft Terminal Server, Citrix Metaframe, etc.), application virtualization server (Microsoft App-V, Citrix XenApp, VMWare ThinApp, etc.) or virtual machine environment within your internal network for the sole and exclusive purpose of permitting individual users within your organization to access and use the Software through a terminal server, application virtualization session, or virtual machine environment from another computer provided that the total number of users that access or use the ADEE Software concurrently at any given point in time on such network, virtual machine or terminal server does not exceed the Permitted Number; and provided that the total number of users authorized to use the ADEE Software through the terminal server, application virtualization session, or virtual machine environment does not exceed six (6) times the Permitted Number of users. Accordingly, the limitations set forth in Section 1(a)(ii) regarding the number of installations and the requirement that the usage be on the same physical network shall not apply to terminal server, application virtualization session, or virtual machine environments. In a virtual environment, you must deploy a means of preventing users from exceeding the Permitted Number of concurrent users. Altova makes no warranties or representations about the performance of

Altova software in a terminal server, application virtualization session, or virtual machine environment and the foregoing are expressly excluded from the limited warranty in Section 3 hereof and technical support is not available with respect to issues arising from use in such environments.

(iv) You may make one (1) backup and one (1) archival copy of the ADEE software, provided your backup and archival copies are not installed or used on any computer and further provided that all such copies shall bear the original and unmodified copyright, patent and other intellectual property markings that appear on or in the ADEE software. You may not transfer the rights to a backup or archival copy unless you transfer all rights in the ADEE software as provided in this AEULA. You, as the primary user of the computer on which the ADEE software is installed, may also install the ADEE software on one of your home computers for your use. A copy of the ADEE software may be installed on home computers up to a total of the number of Permitted Users provided that the ADEE software will not be used at the same time on a home computer as the ADEE software is being used on the primary computer. If you are using a Concurrent-User version of the ADEE software for home use, then you may install the software on any compatible computers equal to the number of Permitted Users only.

(b) **Key Codes.** Prior to your purchase and as part of the registration for the thirty (30) -day evaluation period, as applicable, you will receive an evaluation key code. You will receive a purchase key code when you elect to purchase the ADEE software licenses from either Altova GmbH or an authorized reseller. The purchase key code will enable you to activate the software beyond the initial evaluation period. You may not re-license, reproduce or distribute any key code except with the express written permission of Altova.

(c) **Limited Transfer Rights.** You may transfer all your rights to use the ADEE software to another person or legal entity provided that: (i) you also transfer each of this AEULA, the ADEE software and all other software or hardware bundled or pre-installed with the ADEE software, including all copies, updates and prior versions, and all copies of font software converted into other formats, to such person or entity; (ii) you retain no copies, including backups and copies stored on a computer; (iii) the receiving party secures a personalized key code from Altova; and (iv) the receiving party accepts the terms and conditions of this AEULA and any other terms and conditions upon which you legally purchased a license to the ADEE software.

(d) **Applicable AEULA Terms.** The terms and conditions set forth in Sections 1, 3 and 7 apply to the ADEE software.

2. Authentic Browser-Plugin Enterprise Edition (“ABEE”) Software Terms and Conditions

(a) **License Grant and Term.** Upon your acceptance of this AEULA, Altova grants you a non-exclusive, non-transferable limited license, without the right to grant sublicenses, to install and use ABEE software on a per server basis for a twelve (12) month term, commencing on the date of your license purchase and expiring on the date that is twelve (12) months thereafter (the “ABEE License Term”). Altova also grants you a non-exclusive, non-transferable, limited worldwide license, without the right to grant sublicenses, to use software to develop web pages, web applications, or applications that include ABEE software, to reproduce the ABEE software on your website or server and to distribute the ABEE software from your website or server over a computer network, but only in its executable object code form, and only to end users for the limited purpose of enabling them to view, share, and/or edit XML files during the ABEE License Term. If you wish to continue to use, and/or reproduce and/or distribute the ABEE software after the expiration of its license term, you must purchase a new Authentic Browser-Plugin Enterprise Edition. If you have purchased an ABEE software license then under the terms of the AEULA, support and maintenance (or SMP as further detailed below) for the software is included as part of the license

purchase and you will be entitled to receive the benefits set forth below during the ABEE License Term which is coterminous with the Support Period. Unlike other Altova software products, you cannot renew SMP for the ABEE software and at the expiration of the ABEE License Term and Support Period, you must purchase a new ABEE software license if you wish to continue to use, reproduce or distribute the ABEE software.

(b) Key Codes. Prior to your purchase you may request a thirty (30) day evaluation key code, which will be sent to you. **You will receive a purchase key code when you elect to purchase the ABEE software licenses from either Altova GmbH or an authorized reseller. The purchase key code will enable you to use the ABEE software during the ABEE License Term. You may not re-license, reproduce or distribute any key code except with the express written permission of Altova.**

(c) ABEE Software Specific Restrictions. In addition to the restrictions and obligations provided in other sections of this AEULA that are applicable to the ABEE software, your limited license to distribute the ABEE software set forth above, is further subject to all of the following restrictions; (i) ABEE software may only be licensed but may not be sold, and (ii) you must use, reproduce or distribute the ABEE software provided by Altova **AS IS** and may not impair, alter or remove Altova's AEULA, (which will appear in the installation process and which an end user must accept in order to be able to install or operate the software or any other files).

(d) Applicable AEULA Terms. The terms and conditions set forth in Sections 2, 3 and 7 apply to the ABEE software.

3. Authentic Enterprise Editions (ADEE and ABEE) Software Additional Terms and Conditions

The terms set forth in Section 3 are applicable to the ADEE and ABEE software licenses and are in addition to the specific terms applicable to those software licenses.

(a) Upgrades and Updates. If the software that you have licensed is an upgrade or an update, then the latest update or upgrade that you download and install replaces all or part of the ADEE or ABEE software previously licensed. The update or upgrade and the associated license keys, as applicable, does not constitute the granting of a second license to the software in that you may not use the upgrade or updated copy in addition to the copy of the software that it is replacing and whose license has terminated.

(b) Support and Maintenance. Altova offers "Support & Maintenance Package(s)" ("SMP") for the ADEE and ABEE software product editions that you have licensed. The Support Period, hereinafter defined, covered by such SMP shall be delineated at such time as you elect to purchase a SMP. In the case of your ABEE software license, twelve months of SMP is included that is coterminous with the ABEE License Term. Your rights with respect to support and maintenance as well as your upgrade eligibility depend on your decision to purchase SMP and the level of SMP that you have purchased:

(i) If you have not purchased SMP, you will receive the software AS IS and will not receive any maintenance releases or updates. However, Altova, at its option and in its sole discretion on a case by case basis, may decide to offer maintenance releases to you as a courtesy, but these maintenance releases will not include any new features in excess of the feature set at the time of your purchase of the Software. In addition, Altova will provide free technical support to you for thirty (30) days after the date of your purchase (the "Support Period" for the purposes of this paragraph 3(b)), and Altova, in its sole discretion on a case by case basis, may also provide free courtesy technical support during your thirty (30) -day evaluation period.

Technical support is provided via a Web-based support form only, and there is no guaranteed response time.

(ii) If you have purchased SMP then, solely for the duration of its delineated Support Period, **you are eligible to receive the version of the ADEE or ABEE software edition** that you have licensed and all maintenance releases and updates for that edition that are released during your Support Period. For the duration of your SMP's Support Period, you will also be eligible to receive upgrades to the comparable edition of the next version of the ADEE or ABEE software that succeeds the software edition that you have licensed for applicable upgrades released during your Support Period. The specific upgrade edition that you are eligible to receive based on your Support Period is further detailed in the SMP that you have purchased. Software that is introduced as a separate product is not included in SMP. Maintenance releases, updates and upgrades may or may not include additional features. In addition, Altova will provide Priority Technical Support to you for the duration of the Support Period. Priority Technical Support is provided via a Web-based support form only and Altova will make commercially reasonable efforts to respond via e-mail to all requests within forty-eight (48) hours during Altova's business hours (MO-FR, 8am UTC – 10pm UTC, Austrian and US holidays excluded) and to make reasonable efforts to provide work-arounds to errors reported in the software.

(iii) During the Support Period you may also report any software problem or error to Altova. If Altova determines that a reported reproducible material error in the software exists and significantly impairs the usability and utility of the ADEE or ABEE software, Altova agrees to use reasonable commercial efforts to correct or provide a usable work-around solution in an upcoming maintenance release or update, which is made available at certain times at Altova's sole discretion. If Altova, in its discretion, requests written verification of an error or malfunction discovered by you or requests supporting example files that exhibit the software problem, you shall promptly provide such verification or files, by email, telecopy, or overnight mail, setting forth in reasonable detail the respects in which the ADEE or ABEE software fails to perform. You shall use reasonable efforts to cooperate in diagnosis or study of errors. Altova may include error corrections in maintenance releases, updates, or new major releases of the software. Altova is not obligated to fix errors that are immaterial. Immortal errors are those that do not significantly impact use of the software as determined by Altova in its sole discretion. Whether or not you have purchased the Support & Maintenance Package, technical support only covers issues or questions resulting directly out of the operation of the ADEE or ABEE software and Altova will not provide you with generic consultation, assistance, or advice under any circumstances.

(iv) Updating the ADEE or ABEE software may require the updating of software not covered by this AEULA before installation. Updates of the operating system and application software not specifically covered by this AEULA are your responsibility and will not be provided by Altova under this AEULA. Altova's obligations under this Section are contingent upon your proper use of the ADEE or ABEE software and your compliance with the terms and conditions of this AEULA at all times. Altova shall be under no obligation to provide the above technical support if, in Altova's opinion, the ADEE or ABEE software has failed due to the following conditions: (a) damage caused by the relocation of the software to another location or CPU; (b) alterations, modifications or attempts to change the software without Altova's written approval; (c) causes external to the software, such as natural disasters, the failure or fluctuation of electrical power, or computer equipment failure; (d) your failure to maintain the software at Altova's specified release level; or (e) use of the software with other software without Altova's prior written approval. It will be your sole responsibility to: (a) comply with all Altova-specified operating and troubleshooting procedures and then notify Altova immediately of ADEE or ABEE software malfunction and provide Altova with complete information thereof; (b) provide for the security of your confidential information; and (c) establish and maintain backup systems and procedures necessary to reconstruct lost or altered files, data or programs.

(c) Limited Warranty. Altova warrants to the person or entity that first purchases a license for use of the ADEE or ABEE software pursuant to the terms of this AEULA that (i) the software will perform substantially in accordance with any accompanying documentation for a period of ninety (90) days from the date of receipt, and (ii) any support services provided by Altova shall be substantially as described in Section 3(b) of this AEULA. Some states and jurisdictions do not allow limitations on duration of an implied warranty, so the above limitation may not apply to you. To the extent allowed by applicable law, implied warranties on the ADEE or ABEE software, if any, are limited to ninety (90) days. Altova's and its suppliers' entire liability and your exclusive remedy shall be, at Altova's option, either (i) return of the price paid, if any, or (ii) repair or replacement of the ADEE or ABEE software that does not meet Altova's Limited Warranty and which is returned to Altova with a copy of your receipt. This Limited Warranty is void if failure of the ADEE or ABEE software has resulted from accident, abuse, misapplication, abnormal use, Trojan horse, virus, or any other malicious external code. Any replacement ADEE or ABEE software will be warranted for the remainder of the original warranty period or thirty (30) days, whichever is longer. This limited warranty does not apply to Evaluation Software. THE FOREGOING LIMITED WARRANTY AND REMEDIES STATE THE SOLE AND EXCLUSIVE REMEDIES FOR ALTOVA'S OR ITS SUPPLIERS' BREACH OF WARRANTY. ALTOVA AND ITS SUPPLIERS DO NOT AND CANNOT WARRANT THE PERFORMANCE OR RESULTS YOU MAY OBTAIN BY USING THE SOFTWARE. EXCEPT FOR THE FOREGOING LIMITED WARRANTY, AND FOR ANY WARRANTY, CONDITION, REPRESENTATION OR TERM TO THE EXTENT WHICH THE SAME CANNOT OR MAY NOT BE EXCLUDED OR LIMITED BY LAW APPLICABLE TO YOU IN YOUR JURISDICTION, ALTOVA AND ITS SUPPLIERS MAKE NO WARRANTIES, CONDITIONS, REPRESENTATIONS OR TERMS, EXPRESS OR IMPLIED, WHETHER BY STATUTE, COMMON LAW, CUSTOM, USAGE OR OTHERWISE AS TO ANY OTHER MATTERS. TO THE MAXIMUM EXTENT PERMITTED BY APPLICABLE LAW, ALTOVA AND ITS SUPPLIERS DISCLAIM ALL OTHER WARRANTIES AND CONDITIONS, EITHER EXPRESS OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, IMPLIED WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE, SATISFACTORY QUALITY, INFORMATIONAL CONTENT OR ACCURACY, QUIET ENJOYMENT, TITLE AND NON-INFRINGEMENT, WITH REGARD TO THE SOFTWARE, AND THE PROVISION OF OR FAILURE TO PROVIDE SUPPORT SERVICES. THIS LIMITED WARRANTY GIVES YOU SPECIFIC LEGAL RIGHTS. YOU MAY HAVE OTHERS, WHICH VARY FROM STATE/JURISDICTION TO STATE/JURISDICTION.

(d) Limitation of Liability and Infringement Claims. TO THE MAXIMUM EXTENT PERMITTED BY APPLICABLE LAW EVEN IF A REMEDY FAILS ITS ESSENTIAL PURPOSE, IN NO EVENT SHALL ALTOVA OR ITS SUPPLIERS BE LIABLE FOR ANY SPECIAL, INCIDENTAL, DIRECT, INDIRECT OR CONSEQUENTIAL DAMAGES WHATSOEVER (INCLUDING, WITHOUT LIMITATION, DAMAGES FOR LOSS OF BUSINESS PROFITS, BUSINESS INTERRUPTION, LOSS OF BUSINESS INFORMATION, OR ANY OTHER PECUNIARY LOSS) ARISING OUT OF THE USE OF OR INABILITY TO USE THE ADEE or ABEE SOFTWARE OR THE PROVISION OF OR FAILURE TO PROVIDE SUPPORT SERVICES, EVEN IF ALTOVA HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES. IN ANY CASE, ALTOVA'S ENTIRE LIABILITY UNDER ANY PROVISION OF THIS AEULA SHALL BE LIMITED TO THE AMOUNT ACTUALLY PAID BY YOU FOR THE ADEE OR ABEE SOFTWARE PRODUCT. Because some states and jurisdictions do not allow the exclusion or limitation of liability, the above limitation may not apply to you. In such states and jurisdictions, Altova's liability shall be limited to the greatest extent permitted by law and the limitations or exclusions of warranties and liability contained herein do not prejudice applicable statutory consumer rights of person acquiring goods otherwise than in the course of business. The disclaimer and limited liability above are fundamental to this AEULA between Altova and you. Altova will indemnify and hold you harmless and will defend or settle any claim, suit or proceeding

brought against you by a third party that is based upon a claim that the content contained in the ADEE or ABEE software infringes a copyright or violates an intellectual or proprietary right protected by United States or European Union law ("Claim"), but only to the extent the Claim arises directly out of the use of the Software and subject to the limitations set forth in this Section 3(d) of this Agreement except as otherwise expressly provided. You must notify Altova in writing of any Claim within ten (10) business days after you first receive notice of the Claim, and you shall provide to Altova at no cost such assistance and cooperation as Altova may reasonably request from time to time in connection with the defense of the Claim. Altova shall have sole control over any Claim (including, without limitation, the selection of counsel and the right to settle on your behalf on any terms Altova deems desirable in the sole exercise of its discretion). You may, at your sole cost, retain separate counsel and participate in the defense or settlement negotiations. Altova shall pay actual damages, costs, and attorney fees awarded against you (or payable by you pursuant to a settlement agreement) in connection with a Claim to the extent such direct damages and costs are not reimbursed to you by insurance or a third party, to an aggregate maximum equal to the purchase price of the ADEE or ABEE software. If the ADEE or ABEE software or its use becomes the subject of a Claim or its use is enjoined, or if in the opinion of Altova's legal counsel the software is likely to become the subject of a Claim, Altova shall attempt to resolve the Claim by using commercially reasonable efforts to modify the ADEE or ABEE software or obtain a license to continue using the ADEE or ABEE software. If in the opinion of Altova's legal counsel the Claim, the injunction or potential Claim cannot be resolved through reasonable modification or licensing, Altova, at its own election, may terminate this AEULA without penalty, and will refund to you on a pro rata basis any fees paid in advance by you to Altova. **THE FOREGOING CONSTITUTES ALTOVA'S SOLE AND EXCLUSIVE LIABILITY FOR INTELLECTUAL PROPERTY INFRINGEMENT.** This indemnity does not apply to infringements that would not be such, except for customer-supplied elements.

(e) Evaluation Software. This section applies to all evaluation copies of the ADEE or ABEE software ("Evaluation Software") and continues in effect until you purchase a license. **THE EVALUATION SOFTWARE IS PROVIDED TO YOU "AS-IS" WITH NO WARRANTIES FOR USE OR PERFORMANCE, AND ALTOVA DISCLAIMS ANY WARRANTY OR LIABILITY OBLIGATIONS TO YOU OF ANY KIND, WHETHER EXPRESS OR IMPLIED.** WHERE LEGALLY LIABILITY CANNOT BE EXCLUDED FOR PRE-RELEASE AND/OR EVALUATION SOFTWARE, BUT IT MAY BE LIMITED, ALTOVA'S LIABILITY AND THAT OF ITS SUPPLIERS SHALL BE LIMITED TO THE SUM OF FIFTY DOLLARS (USD \$50) IN TOTAL. If the Evaluation Software has a time-out feature, then the software will cease operation after the conclusion of the designated evaluation period. Access to any files created with the Evaluation Software is entirely at your risk.

4. Authentic Desktop Community Edition ("ADCE") Software Terms and Conditions

(a) License Grant and Keycode. Upon your acceptance of this AEULA, Altova grants you a non-exclusive, non-transferable limited license, without the right to grant sublicenses, to install and use a copy of ADCE software on your compatible personal computer or workstation for the purpose of viewing, distributing, sharing, and editing of XML files solely in connection with STYLEVISION® Power Stylesheets as further provided herein. You will receive a key code that will enable you to activate or operate the ADCE software. You may not re-license, reproduce or distribute any key code except with the express written permission of Altova. You may make one (1) backup and one (1) archival copy of the ADCE software, provided your backup and archival copies are not installed or used on any computer and further provided that all such copies shall bear the original and unmodified copyright, patent and other intellectual property markings that appear on or in the ADCE. You may install one (1) copy of such Setup Program for ADCE software on a computer file server within your internal network for the sole and exclusive purpose of installing the ADCE software to an unlimited number of client computers on your internal network. No other server or network use of the ADCE software is permitted, including but not

limited to using the ADCE software (i) either directly or through commands, data or instructions from or to another computer or (ii) for internal network, internet or web hosting services.

(b) Distribution. Upon your acceptance of this AEULA as part of your use of the ADCE software, and subject to your ongoing compliance with its terms and conditions, Altova hereby grants ADCE software users a non-exclusive, non-transferable, limited license, without the right to grant sublicenses, to reproduce the Setup Program for the ADCE software and distribute the Setup Program for the ADCE software in executable form to end users in the manner hereinafter provided. You may distribute the Setup Program for the ADCE software to any third party electronically or via download from the website or on physical media such as CD-ROMS or diskettes as part of or in conjunction with products that you have developed.

(c) ADCE Software Specific Restrictions. In addition to the restrictions and obligations provided in other sections of this AEULA, your license to distribute the Setup Program for the ACDE software is further subject to all of the following restrictions: (i) ACDE software shall only be licensed and not sold, (ii) you may not make the ACDE software available as a stand-alone product and if distributed as part of a product bundle you may charge for the product bundle provided that you license such product bundle at the same or lower fee at which you license any reasonably equivalent product bundle which does not include the ACDE software, (iii) you must use the Setup Program for ACDE provided by Altova AS IS and may not impair, alter or remove Altova's AEULA, (which will appear in the installation process and which an end user must accept in order to be able to install or operate the ACDE software) or any other files, and (iv) you may not combine the ACDE software with your product in such a way that your product modifies or generates Stylevision Power Stylesheets.

(d) Applicable AEULA Terms. The terms and conditions set forth in Sections 4, 6 and 7 apply to the ADCE software.

5. Authentic Browser-Plugin Community Edition (“ABCE”) Software Terms and Conditions

(a) License Grant and Distribution. Upon your acceptance of this AEULA, Altova grants you a non-exclusive, non-transferable limited license, without the right to grant sublicenses, to use and develop web pages, web applications, or applications that include the ABCE software, to reproduce the ABCE software and to distribute the ABCE software in executable form in the manner hereinafter provided to end users for the purpose of viewing, sharing and editing XML files solely in connection with StyleVision® Power Stylesheets as further provided herein. You may install the ABCE software on a web server within your network for the purpose of downloading and installing the ABCE software (to an unlimited number of client computers on your internal network). You may distribute the ABCE software to any third party electronically or via download from the website or on physical media such as CD-ROMS or diskettes as part of or in conjunction with products that you have developed.

(b) ABCE Software Specific Restrictions. In addition to the restrictions and obligations provided in other sections of this AEULA, your license to distribute ABCE software is further subject to all of the following restrictions: (i) the ABCE software shall only be licensed and not sold; (ii) you may not make the ABCE software available as a stand-alone product and if distributed as part of a product bundle you may charge for the product bundle provided that you license such product bundle at the same or lower fee at which you license any reasonably equivalent product bundle which does not include the ABCE software; (iii) you must use the ABCE software provided by Altova AS IS and may not impair, alter or remove Altova's AEULA (which will appear in the installation process and which an end user must accept in order to be able to install or operate the ABCE software) or any other files; (iv) other Altova products cannot

be distributed under this AEULA; and (v) you may not combine the ABCE software with your product in such a way that your product modifies or generates Stylevision Power Stylesheet(s).

(c) Applicable AEULA Terms. The terms and conditions set forth in Sections 5, 6 and 7 apply to the ABCE software.

6. Authentic Community Editions (ADCE and ABCE) Software Additional Terms and Conditions

The terms set forth in Section 6 are applicable to the ADCE and ABCE software licenses and are in addition to the specific terms applicable to those software licenses.

(a) Use Limitation. The ADCE and ABCE software are licensed and distributed by Altova for viewing, distributing, sharing, and editing of XML files solely in connection with StyleVision® Power Stylesheets, defined as .sps files that are template files developed by Altova or its customers using Altova's StyleVision® product. You are not authorized to integrate or use the ADCE or ABCE software with (i) any StyleVision® Power Stylesheet(s) not developed in accordance with the Altova Software License Agreement available at <http://www.altova.com/eula> or (ii) other software or enhancement that uses Inter Application Communication (IAC) to programmatically interface with ADCE or ABCE software for the purpose of enabling additional functionality normally not available in ADCE or ABCE software or providing functionality that competes with other Altova products.

(b) Warranty Disclaimer. THE ADCE OR ABCE SOFTWARE IS PROVIDED TO YOU FREE OF CHARGE, AND ON AN "AS-IS" BASIS. ALTOVA PROVIDES NO WARRANTIES FOR THE ADCE OR ABCE SOFTWARE. TO THE MAXIMUM EXTENT PERMITTED BY LAW, ALTOVA AND ITS SUPPLIERS DISCLAIM ALL WARRANTIES AND REPRESENTATIONS, WHETHER EXPRESS, IMPLIED OR OTHERWISE, INCLUDING WITHOUT LIMITATION ANY IMPLIED WARRANTIES OF FITNESS FOR A PARTICULAR PURPOSE, MERCHANTABILITY, SATISFACTORY QUALITY, INFORMATIONAL CONTENT, OR ACCURACY, QUIET ENJOYMENT, TITLE, AND NON-INFRINGEMENT. ALTOVA DOES NOT WARRANT THAT THE ADCE OR ABCE SOFTWARE IS ERROR-FREE OR WILL OPERATE WITHOUT INTERRUPTION. IF MANDATORILY APPLICABLE LAW REQUIRES ANY WARRANTIES WITH RESPECT TO THE ADCE OR ABCE SOFTWARE, ALL SUCH WARRANTIES ARE LIMITED IN DURATION TO THIRTY (30) DAYS FROM THE DATE OF INSTALLATION OR BEGIN OF USE, WHATEVER IS THE EARLIER. SOME STATES OR JURISDICTIONS DO NOT ALLOW THE EXCLUSION OF IMPLIED WARRANTIES, SO THE ABOVE EXCLUSION MAY NOT APPLY TO YOU. THIS WARRANTY GIVES YOU SPECIFIC LEGAL RIGHTS AND YOU MAY ALSO HAVE OTHER LEGAL RIGHTS THAT VARY FROM STATE TO STATE OR FROM JURISDICTION TO JURISDICTION. YOU AGREE THAT YOU ARE SOLELY RESPONSIBLE FOR THE ACCURACY AND ADEQUACY OF THE ADCE OR ABCE SOFTWARE FOR YOUR INTENDED USE AND YOU WILL INDEMNIFY AND HOLD HARMLESS ALTOVA FROM ANY 3RD PARTY SUIT TO THE EXTENT BASED UPON THE ACCURACY AND ADEQUACY OF THE ADCE OR ABCE SOFTWARE IN YOUR USE.

(c) Limitation of Liability. TO THE MAXIMUM EXTENT PERMITTED BY APPLICABLE LAW, IN NO EVENT SHALL ALTOVA OR ITS SUPPLIERS BE LIABLE FOR ANY SPECIAL, INCIDENTAL, DIRECT, INDIRECT OR CONSEQUENTIAL DAMAGES WHATSOEVER (INCLUDING, WITHOUT LIMITATION, DAMAGES FOR LOSS OF BUSINESS PROFITS, BUSINESS INTERRUPTION, LOSS OF BUSINESS INFORMATION, OR ANY OTHER PECUNIARY LOSS) ARISING OUT OF THE USE OF OR INABILITY TO USE THE AUTHENTIC SOFTWARE, THE PROVISION OF OR FAILURE TO PROVIDE SUPPORT SERVICES, OR ANY PROVISION OF THIS AEULA, EVEN IF ALTOVA HAS BEEN ADVISED OF THE POSSIBILITY

OF SUCH DAMAGES. WHERE LEGALLY, LIABILITY CANNOT BE EXCLUDED, BUT MAY BE LIMITED, ALTOVA'S LIABILITY AND THAT OF ITS SUPPLIERS SHALL BE LIMITED TO THE SUM OF FIFTY DOLLARS (USD \$50) IN TOTAL. BECAUSE SOME STATES AND JURISDICTIONS DO NOT ALLOW THE EXCLUSION OR LIMITATION OF LIABILITY, THE ABOVE LIMITATION MAY NOT APPLY TO YOU. IN SUCH STATES AND JURISDICTIONS, ALTOVA'S LIABILITY AND THAT OF ITS SUPPLIERS SHALL BE LIMITED TO THE GREATEST EXTENT PERMITTED BY LAW. THE FOREGOING LIMITATIONS ON LIABILITY ARE INTENDED TO APPLY TO THE WARRANTIES AND DISCLAIMERS ABOVE AND ALL OTHER ASPECTS OF THIS AEULA.

(d) Support. Altova is not obliged to provide technical support with respect to the ADCE or ABCE software, that is provided on an AS IS basis. To the extent that Altova in its sole discretion does provide support for these products, the technical support will be provided in the manner detailed in Section 3(b) of this AEULA and subject to all requirements therein contained.

7. Authentic Software (ADEE, ABEE, ADCE and ABCE) Additional Terms and Conditions

The terms set forth in Section 7 are applicable to the ADEE, ABEE, ADCE and ABCE software licenses and are in addition to the specific terms applicable to those software licenses.

(a) Title. Title to the Authentic Software is not transferred to you. Ownership of all copies of the Authentic Software and of copies made by you is vested in Altova, subject to the rights of use or distribution, as applicable, granted to you in this AEULA. All rights not specifically granted in this AEULA are reserved by Altova.

(b) Acknowledgement of Altova's Rights. You acknowledge that the Authentic Software and any copies that you are authorized by Altova to make are the intellectual property of and are owned by Altova and its suppliers. The structure, organization and code of the Authentic Software are the valuable trade secrets and confidential information of Altova and its suppliers. The Authentic Software is protected by copyright, including without limitation by United States Copyright Law, international treaty provisions and applicable laws in the country in which it is being used. You acknowledge that Altova retains the ownership of all patents, copyrights, trade secrets, trademarks and other intellectual property rights pertaining to the Authentic Software, and that Altova's ownership rights extend to any images, photographs, animations, videos, audio, music, text and "applets" incorporated into the Authentic Software and all accompanying printed materials. You will take no actions which adversely affect Altova's intellectual property rights in the Authentic Software. Trademarks shall be used in accordance with accepted trademark practice, including identification of trademark owners' names. Trademarks may only be used to identify printed output produced by the Authentic Software, and such use of any trademark does not give you any right of ownership in that trademark. XMLSpy®, Authentic®, StyleVision®, MapForce®, UModel®, DatabaseSpy®, DiffDog®, SchemaAgent®, SemanticWorks®, MissionKit®, Markup Your Mind®, Nanonull™, and Altova® are trademarks of Altova GmbH (registered in numerous countries). Unicode and the Unicode Logo are trademarks of Unicode, Inc. Windows, Windows XP, Windows Vista, and Windows 7 are trademarks of Microsoft. W3C, CSS, DOM, MathML, RDF, XHTML, XML and XSL are trademarks (registered in numerous countries) of the World Wide Web Consortium (W3C); marks of the W3C are registered and held by its host institutions, MIT, INRIA and Keio. Except as expressly stated above, this AEULA does not grant you any intellectual property rights in the Authentic Software. Notifications of claimed copyright infringement should be sent to Altova's copyright agent as further provided on the Altova Web Site.

(c) Common Restrictions.

(i) You may not reverse engineer, decompile, disassemble or otherwise attempt to discover the source code, underlying ideas, underlying user interface techniques or algorithms of the Authentic Software by any means whatsoever, directly or indirectly, or disclose any of the foregoing, except to the extent you may be expressly permitted to decompile under applicable law in the European Union, if it is essential to do so in order to achieve operability of the Authentic Software with another software program, and you have first requested Altova to provide the information necessary to achieve such operability and Altova has not made such information available. Altova has the right to impose reasonable conditions and to request a reasonable fee before providing such information. Any information supplied by Altova or obtained by you, as permitted hereunder, may only be used by you for the purpose described herein and may not be disclosed to any third party or used to create any software which is substantially similar to the expression of the Authentic Software. Requests for information from users in the European Union with respect to the above should be directed to the Altova Customer Support Department. You may not loan, rent, lease, sublicense, distribute or otherwise transfer all or any portion of the Authentic Software to third parties except to the limited extent expressly provided in this AEULA.

(ii) You may not copy, distribute, or make derivative works of the Authentic Software except as expressly set forth above, and any copies that you are permitted to make pursuant to this Authentic EULA must contain the same copyright, patent and other intellectual property markings that appear on or in the Authentic Software. You may not modify, adapt or translate the Authentic Software. You may not, directly or indirectly, encumber or suffer to exist any lien or security interest on the Authentic Software; knowingly take any action that would cause the Authentic Software to be placed in the public domain; or use the Authentic Software in any computer environment not specified in this Authentic EULA. You will comply with applicable law and Altova's instructions regarding the use of the Authentic Software. You agree to notify your employees and agents who may have access to the Authentic Software of the restrictions contained in this AEULA and to ensure their compliance with these restrictions. You may not alter or modify the Authentic Software or create a new installer for the Authentic Software.

(d) Authentic Software Activation, Updates, Metering and Data Use.

(i) Altova has a built-in license metering module that helps you to avoid any unintentional violation of this AEULA. Altova may use your internal network for license metering between installed versions of the Authentic Software. **Altova's Authentic Software may use your internal network and Internet connection for the purpose of transmitting license-related data at the time of installation, registration, use, or update to an Altova-operated license server and validating the authenticity of the license-related data in order to protect Altova against unlicensed or illegal use of the Authentic Software and to improve customer service. Activation is based on the exchange of license related data between your computer and the Altova license server. You agree that Altova may use these measures and you agree to follow any applicable requirements. You further agree that use of license key codes that are not or were not generated by Altova and lawfully obtained from Altova or an authorized reseller as part of an effort to activate or use the Authentic Software violates Altova's intellectual property rights as well as the terms of this AEULA. You agree that efforts to circumvent or disable Altova's copyright protection mechanisms or license management mechanism violate Altova's intellectual property rights as well as the terms of this AEULA. Altova expressly reserves the rights to seek all available legal and equitable remedies to prevent such actions and to recover lost profits, damages and costs.**

(ii) Altova provides a new LiveUpdate notification service to you, which is free of charge. Altova may use your internal network and Internet connection for the purpose of transmitting license-related data to an Altova-operated LiveUpdate server to validate your license

at appropriate intervals and determine if there is any update available for you. The terms and conditions of the Privacy Policy are set out in full at <http://www.altova.com/privacy> and are incorporated by reference into this AEULA. By your acceptance of the terms of this AEULA or use of the Authentic Software, you authorize the collection, use and disclosure of information collected by Altova for the purposes provided for in this AEULA and/or the Privacy Policy.. Altova has the right in its sole discretion to amend this provision of the AEULA and/or Privacy Policy at any time. You are encouraged to review the terms of the Privacy Policy as posted on the Altova Web site from time to time.

(iii) **Notice to European Users.** Please note that the information as described in paragraph 7(d) above may be transferred outside of the European Economic Area, for purposes of processing, analysis, and review, by Altova, Inc. a company located in Beverly, Massachusetts, U.S.A., or its subsidiaries or Altova's subsidiaries or divisions, or authorized partners, located worldwide. You are advised that the United States uses a sectoral model of privacy protection that relies on a mix of legislation, governmental regulation, and self-regulation. You are further advised that the Council of the European Union has found that this model does not provide "adequate" privacy protections as contemplated by Article 25 of the European Union's Data Directive. (Directive 95/46/EC, 1995 O.J. (L 281) 31). Article 26 of the European Union's Data Directive allows for transfer of personal data from the European Union to a third country if the individual has unambiguously given his consent to the transfer of personal information, regardless of the third country's level of protection. By agreeing to this AEULA, you consent to the transfer of all such information to the United States and the processing of that information as described in this AEULA and the Privacy Policy.

(e) Disclaimer. THE AUTHENTIC SOFTWARE IS NEITHER GUARANTEED NOR WARRANTED TO BE ERROR-FREE NOR SHALL ANY LIABILITY BE ASSUMED BY ALTOVA IN THIS RESPECT. NOTWITHSTANDING ANY SUPPORT FOR ANY TECHNICAL STANDARD, THE AUTHENTIC SOFTWARE IS NOT INTENDED FOR USE IN OR IN CONNECTION WITH, WITHOUT LIMITATION, THE OPERATION OF NUCLEAR FACILITIES, AIRCRAFT NAVIGATION, COMMUNICATION SYSTEMS, AIR TRAFFIC CONTROL EQUIPMENT, MEDICAL DEVICES OR LIFE SUPPORT SYSTEMS, MEDICAL OR HEALTH CARE APPLICATIONS, OR OTHER APPLICATIONS WHERE THE FAILURE OF THE AUTHENTIC SOFTWARE OR ERRORS IN DATA PROCESSING COULD LEAD TO DEATH, PERSONAL INJURY OR SEVERE PHYSICAL OR ENVIRONMENTAL DAMAGE. YOU AGREE THAT YOU ARE SOLELY RESPONSIBLE FOR THE ACCURACY AND ADEQUACY OF THE AUTHENTIC SOFTWARE AND ANY DATA GENERATED OR PROCESSED BY THE SOFTWARE FOR YOUR INTENDED USE AND YOU WILL DEFEND, INDEMNIFY AND HOLD ALTOVA, ITS OFFICERS AND EMPLOYEES HARMLESS FROM ANY 3RD PARTY CLAIMS, DEMANDS, OR SUITS THAT ARE BASED UPON THE ACCURACY AND ADEQUACY OF THE AUTHENTIC SOFTWARE IN YOUR USE OR ANY DATA GENERATED BY THE AUTHENTIC SOFTWARE IN YOUR USE.

(f) Restricted Rights Notice and Export Restrictions. The Authentic Software was developed entirely at private expense and is commercial computer software provided with RESTRICTED RIGHTS. Use, duplication or disclosure by the U.S. Government or a U.S. Government contractor or subcontractor is subject to the restrictions set forth in this Agreement and as provided in FAR 12.211 and 12.212 (48 C.F.R. §12.211 and 12.212) or DFARS 227.7202 (48 C.F.R. §227-7202) as applicable. Consistent with the above, as applicable, Commercial Computer Software and Commercial Computer Documentation licensed to U.S. government end users only as commercial items and only with those rights as are granted to all other end users under the terms and conditions set forth in this AEULA. Manufacturer is Altova GmbH, Rudolfsplatz, 13a/9, A-1010 Vienna, Austria/EU. You may not use or otherwise export or re-export the Authentic Software or documentation except as authorized by United States law and the laws of the jurisdiction in which the Authentic Software was obtained. In particular, but without

limitation, the Authentic Software or Documentation may not be exported or re-exported (i) into (or to a national or resident of) any U.S. embargoed country or (ii) to anyone on the U.S. Treasury Department's list of Specially Designated Nationals or the U.S. Department of Commerce's Table of Denial Orders. By using the Software, you represent and warrant that you are not located in, under control of, or a national or resident of any such country or on any such list.

(g) Termination. Without prejudice to any other rights or remedies of Altova, this AEULA may be terminated (i) by you giving Altova written notice of termination or (ii) by Altova, at its option, giving you written notice of termination or (iii) Altova giving you written notice of termination if you fail to comply with the terms and conditions of the AEULA. This AEULA automatically terminates upon the expiration of the ABEE License Term. Upon any termination or expiration of this AEULA, you must cease all use of Authentic Software, licensed hereunder, destroy all copies then in your possession or control and take such other actions as Altova may reasonably request to ensure that no copies of the Authentic Software remain in your possession or control. The terms and conditions set forth in Sections 1(e), 2(c)-(d), 3(c)-(d), 4(c)-(d), 5(b)-(c), 6(b)-(c) and 7 survive termination of this AEULA as applicable.

(h) Third Party Software. The Authentic Software may contain third party software which requires notices and/or additional terms and conditions. Such required third party software notices and/or additional terms and conditions are located at our website at http://www.altova.com/legal_3rdparty.html and are made a part of and incorporated by reference into this AEULA. By accepting this AEULA, you are also accepting the additional terms and conditions, if any, set forth therein.

(i) General Legal Provisions. This AEULA contains the entire agreement and understanding of the parties with respect to the subject matter hereof, and supersedes all prior written and oral understandings of the parties with respect to the subject matter hereof. Any notice or other communication given under this AEULA shall be in writing and shall have been properly given by either of us to the other if sent by certified or registered mail, return receipt requested, or by overnight courier to the address shown on Altova's Web site for Altova and the address shown in Altova's records for you, or such other address as the parties may designate by notice given in the manner set forth above. This AEULA will bind and inure to the benefit of the parties and our respective heirs, personal and legal representatives, affiliates, successors and permitted assigns. The failure of either of us at any time to require performance of any provision hereof shall in no manner affect such party's right at a later time to enforce the same or any other term of this AEULA. This AEULA may be amended only by a document in writing signed by both of us. In the event of a breach or threatened breach of this AEULA by either party, the other shall have all applicable equitable as well as legal remedies. Each party is duly authorized and empowered to enter into and perform this AEULA. If, for any reason, any provision of this AEULA is held invalid or otherwise unenforceable, such invalidity or unenforceability shall not affect the remainder of this AEULA, and this AEULA shall continue in full force and effect to the fullest extent allowed by law. The parties knowingly and expressly consent to the foregoing terms and conditions.

(i) If you are located in the European Union and are using the Authentic Software in the European Union and not in the United States, then this AEULA will be governed by and construed in accordance with the laws of the Republic of Austria (excluding its conflict of laws principles and the U.N. Convention on Contracts for the International Sale of Goods) and you expressly agree that exclusive jurisdiction for any claim or dispute with Altova or relating in any way to your use of the Authentic Software resides in the Handelsgericht, Wien (Commercial Court, Vienna) and you further agree and expressly consent to the exercise of personal jurisdiction in the Handelsgericht, Wien (Commercial Court, Vienna) in connection with any such dispute or claim.

(ii) If you are located in the United States or are using the Authentic Software in the United States then this AEULA will be governed by and construed in accordance with the laws of the Commonwealth of Massachusetts, USA (excluding its conflict of laws principles and the U.N. Convention on Contracts for the International Sale of Goods) and you expressly agree that exclusive jurisdiction for any claim or dispute with Altova or relating in any way to your use of the Authentic Software resides in the federal or state courts of the Commonwealth of Massachusetts and you further agree and expressly consent to the exercise of personal jurisdiction in the federal or state courts of the Commonwealth of Massachusetts in connection with any such dispute or claim.

(iii) If you are located outside of the European Union or the United States and are not using the Authentic Software in the United States, then this AEULA will be governed by and construed in accordance with the laws of the Republic of Austria (excluding its conflict of laws principles and the U.N. Convention on Contracts for the International Sale of Goods) and you expressly agree that exclusive jurisdiction for any claim or dispute with Altova or relating in any way to your use of the Authentic Software resides in the Handelsgericht, Wien (Commercial Court, Vienna) and you further agree and expressly consent to the exercise of personal jurisdiction in the Handelsgericht Wien (Commercial Court, Vienna) in connection with any such dispute or claim. This AEULA will not be governed by the conflict of law rules of any jurisdiction or the United Nations Convention on Contracts for the International Sale of Goods, the application of which is expressly excluded.

Last Updated: 2011-10-01

Índice

A

- Agentes de escucha de eventos (Firefox),** 69
- Archivo CAB,** 10, 13, 50
- Archivo XPI,** 10, 13, 50
- Archivos SPS basados en BD,**
requisitos para trabajar con Authentic Browser, 17
- Atributo CODEBASE,** 24
- Authentic,** 160
- ApplyTextState, 100
 - attachCallBack, 101
 - ControlInitialized, 108
 - CreateChild, 109
 - CurrentSelection, 110
 - DesignDataLoadObject, 111
 - EditClear, 112
 - EditCopy, 113
 - EditCut, 114
 - EditPaste, 115
 - EditRedo, 116
 - EditSelectAll, 117
 - EditUndo, 118
 - event, 124
 - FindDialog, 125
 - FindNext, 126
 - GetAllAttributes, 127
 - GetAllowedElements, 129
 - GetFileVersion, 131
 - GetNextVisible, 132
 - GetPreviousVisible, 133
 - IsEditClearEnabled, 134
 - IsEditCopyEnabled, 135
 - IsEditCutEnabled, 136
 - IsEditPasteEnabled, 137
 - IsEditRedoEnabled, 138
 - IsEditUndoEnabled, 139
 - IsFindNextEnabled, 140
 - IsRowAppendEnabled, 141
 - IsRowDeleteEnabled, 142
 - IsRowDuplicateEnabled, 143
 - IsRowInsertEnabled, 144
 - IsRowMoveDownEnabled, 145
 - IsRowMoveUpEnabled, 146
 - IsTextStateApplied, 147
 - IsTextStateEnabled, 148
 - LoadXML, 149
 - MarkUpView, 150
 - Print, 152
 - PrintPreview, 153
 - ReplaceDialog, 156
 - Reset, 157
 - RowAppend, 158
 - RowDelete, 159
 - RowInsert, 161
 - RowMoveDown, 162
 - RowMoveUp, 163
 - Save, 164
 - SavePOST, 166
 - SaveXML, 167
 - SchemaLoadObject, 168
 - SelectionChanged, 169
 - SelectionMoveTabOrder, 170
 - SelectionSet, 171
 - StartEditing, 173
 - ValidateDocument, 181
 - validationBadData, 182
 - validationMessage, 183
 - XMLDataLoadObject, 184
 - XMLDataSaveUrl, 185
 - XMLRoot, 186
- Authentic Browser,** 97
- archivo del complemento, 10, 13, 50
 - configuración de red, 8
 - control de eventos, 28
 - descarga de archivo en servidor, 50
 - descarga en servidor, 10, 13
 - ID de clase, 10, 24
 - introducción, 8
 - subrutinas, 28
 - tipos MIME, 10, 35
 - ventajas, 7
 - versión, 24, 35
 - versiones, 10
 - y archivos SPS basados en BD, 17
- Authentic RowDuplicate,** 160
- AuthenticDataTransfer,**
dropEffect, 203
getData, 204
ownDrag, 205
type, 206

AuthenticEvent,

altKey, 208
 altLeft, 209
 button, 210
 cancelBubble, 211
 clientX, 212
 clientY, 213
 ctrlKey, 214
 ctrlLeft, 215
 dataTransfer, 216
 fromElement, 217
 keyCode, 218
 propertyName, 219
 repeat, 220
 returnValue, 221
 shiftKey, 222
 shiftLeft, 223
 srcElement, 224
 type, 225

AuthenticRange, 237

AppendRow, 239
 Application, 240
 CanPerformAction, 241
 CanPerformActionWith, 242
 Close, 243
 CollapsToBegin, 244
 CollapsToEnd, 245
 Copy, 246
 Cut, 247
 Delete, 248
 DeleteRow, 249
 DuplicateRow, 250
 ExpandTo, 252
 FirstTextPosition, 253
 FirstXMLData, 254
 FirstXMLDataOffset, 255
 GetElementAttributeNames, 257
 GetElementAttributeValue, 258
 GetElementHierarchy, 259
 GetEntityNames, 260
 Goto, 262
 GotoNext, 263
 GotoNextCursorPosition, 264
 GotoPrevious, 265
 GotoPreviousCursorPosition, 266
 HasElementAttribute, 267
 InsertEntity, 268
 InsertRow, 269

IsEmpty, 273

IsEqual, 274
 IsInDynamicTable, 276
 LastTextPosition, 281
 LastXMLData, 282
 LastXMLDataOffset, 283
 MoveBegin, 285
 MoveEnd, 286
 MoveRowDown, 288
 MoveRowUp, 287
 Parent, 289
 Paste, 290
 PerformAction, 291
 Select, 292
 SelectNext, 293
 SelectPrevious, 294
 SetElementAttributeValue, 295
 SetFromRange, 297
 Text, 299

AuthenticView, 103, 322, 358

Application, 334
 DocumentBegin, 341
 DocumentEnd, 342
 Goto, 348
 MarkupVisibility, 351
 Parent, 352
 Print, 353
 Redo, 354
 Selection, 355
 Undo, 357
 WholeDocument, 359

Ayudantes de entrada, 78**B****Botones de barras de herramientas,**

cambiar el comportamiento de, 70

Botones de estado del texto, 77**Buscar y reemplazar, 74****C****Configuración de red, 8****Configuración del servidor, 13**

Contenido,

acceder y modificar, 72

Contenido del documento,

acceder y modificar, 72

Contrato de licencia para el usuario final, 446**Control de eventos, 28****Controladores de eventos, 67, 70****Controllnitialized, 67, 108****Corrección ortográfica, 79****O****Objeto AuthenticRange, 72****Objeto Authentic, 72****Objeto AuthenticView, 72****Operaciones de edición, 73****Operaciones en filas de tablas, 75****D****Distribución,**

de productos de software de Altova, 446, 447, 448

DOM,

y XMLData, 90

P**Página HTML,**

introducción, 19

Página HTML para Firefox,

agregar agentes de escucha de eventos, 38

ejemplo básico, 39

Elemento EM BED, 35

introducción, 34

ordenar una tabla, 41

Página HTML para IE,

ejemplo básico, 29

elemento OBJECT, 24

elemento SCRIPT, 28

introducción, 23

ordenar una tabla, 31

Página HTML para IE or Firefox,

archivo de ejemplo, 45

introducción, 44

Paquetes, 79**Período de evaluación,**

de los productos de software de Altova, 446, 447, 448

Puntos de conexión, 67**I****ID de clase, 10, 24****Información legal, 446****Información sobre derechos de autor, 446****Internet Information Service para el servidor, 15****R****Reemplazar, 74****Referencia,**

eventos, 71

Referencia del usuario, 65**Requisitos del sistema, 8****L****Licencia,**

información sobre, 446

Licencia para la edición Enterprise Edition, 21

String, 235
URL, 236

S

selectionchanged, 67, 69
Servicio de explorador para el servidor, 15
Subrutinas, 28

T

Tablas dinámicas, 75
Teclas de acceso rápido, 76
Tipos MIME, 10, 35

X

XMLData, 86
 GetChild, 409
 GetChildKind, 412
 HasChildrenKind, 419
 IsSameNode, 423
 y DOM, 90
XMLSpyLib,
 AuthenticDataTransfer, 202
 AuthenticEvent, 207
 XMLSpyXMLData, 402
XMLSPYPLUGINLib,
 Authentic, 97
 XMLSpyXMLLoadSave, 234
XMLSpyXMLData,
 AppendChild, 403
 EraseAllChildren, 406
 EraseCurrentChild, 408
 GetCurrentChild, 413
 GetFirstChild, 414
 GetNextChild, 416
 HasChildren, 418
 InsertChild, 420
 Kind, 424
 MayHaveChildren, 425
 Name, 426
 Parent, 427
 TextValue, 429
XMLSpyXMLLoadSave,