

IBM Daeja ViewONE Standard
IBM Daeja ViewONE Professional
IBM Daeja ViewONE Virtual



Annotations Module Setup Guide

Version 4.1

IBM Daeja ViewONE Standard
IBM Daeja ViewONE Professional
IBM Daeja ViewONE Virtual



Annotations Module Setup Guide

Version 4.1

Note

Before using this information and the product it supports, read the information in “Notices” on page 51.

This edition applies to Version 4 Release 1 Modification 0 of IBM® Daeja ViewONE Standard Edition (product number 5725-Q02), IBM Daeja ViewONE Professional Edition (product number 5725-Q03), and IBM Daeja ViewONE Virtual (product number 5725-Q04) and to all subsequent releases and modifications until otherwise indicated in new editions.

© **Copyright IBM Corporation 2014.**

US Government Users Restricted Rights – Use, duplication or disclosure restricted by GSA ADP Schedule Contract with IBM Corp.

Contents

<i>ibm.com and related resources</i>	7
How to send your comments	7
Contacting IBM	8
<i>Introduction</i>	9
Pre-requisite Requirements	9
Annotation data is kept separate from image files	10
Server Component	10
Annotations and "Flow Layout" format documents	11
IBM Daeja ViewONE and the Annotations User Interface	12
<i>Deployment</i>	13
<i>Configuring the Annotations Server Object</i>	14
Servlet Approach (recommended method)	14
HTTP:POST Approach	15
<i>Customizing the Annotations Module</i>	16
<i>Annotations File Format</i>	17
General Structure	17
EMPTY Marker	18
ANNOTINFO Marker	18
Standard mandatory properties	19
page	19
edit	19
Standard optional properties	20
lineWidth	20
color	20
label	20
tooltip	20
createDate	21
modifiedDate	21
view	21
pageSize	22
pageURL	22
createdId	22
modifiedId	23
blankOutImage	23
customProperty	23
hyperlink	24
Using the IBM Daeja ViewONE annotation tooltips to help with hyperlinks	24
Page hyperlinks	25
Annotation hyperlinks	25
JavaScript hyperlinks	26

Web page hyperlinks	26
hyperlinkSettings	27
Annotation types and their properties.....	29
Arrow	29
Custom	30
Freehand	31
Highlight Rectangle	32
Highlight Polygon.....	33
Line	34
Note	35
Open Polygon	36
Oval	37
Polygon	38
Rectangle	39
Redaction	40
Redaction Polygon.....	41
Stamp	42
Text	43
Appendix A	45
IBM Daeja ViewONE Color Scheme Text.....	45
Appendix B	46
IBM Daeja ViewONE Default Stamps	46
Appendix C	47
IBM Daeja ViewONE Default Web Hyperlink Targets	47
Appendix D.....	48
Annotation Security	48
Simple Annotation Security	48
Extended Annotation Security	49
Notices	51
Trademarks.....	53
Privacy policy considerations	53

ibm.com and related resources

Product support and documentation are available from ibm.com[®].

Support and assistance

Product support is available on the web. Click Support from the product website at:

IBM[®] Daeja[®] ViewONE[®]

http://www.ibm.com/support/entry/portal/product/enterprise_content_management/daeja_viewone

PDF publication

You can view the PDF files online by using the Adobe Acrobat Reader for your operating system. If you do not have the Acrobat Reader installed, you can download it from the Adobe website at <http://www.adobe.com>.

See the following PDF publication website:

Product	Website
IBM Daeja ViewONE Standard	http://www.ibm.com/support/docview.wss?uid=swg27041437
IBM Daeja ViewONE Professional	
IBM Daeja ViewONE Virtual	

How to send your comments

Your feedback helps IBM to provide quality information. Please share any comments that you have about this information or other documentation that IBM Software Development ships with its products.

You can use any of the following methods to provide comments:

- Send your comments by using the online readers' comment form at <http://www.ibm.com/software/data/rcf/>.
- Send your comments by email to comments@us.ibm.com. Include the name of the product, the version number of the product, and the name and publication number of the information (if applicable). If you are commenting on specific text, please include the location of the text (for example, a title, a table number, or a page number).

Contacting IBM

To contact IBM customer service in the United States or Canada, call 1-800-IBM-SERV (1-800-426-7378).

To learn about available service options, call one of the following numbers:

- In the United States: 1-888-426-4343
- In Canada: 1-800-465-9600

For more information about how to contact IBM, see the Contact IBM website at <http://www.ibm.com/contact/us/>.

Introduction

IBM® Daeja® ViewONE® is a Java viewer that extends your web browser so that you can view, zoom, magnify, scroll, pan, rotate and print your images and image documents quickly and easily.

This document describes how to deploy and make use of the IBM Daeja ViewONE Annotations Module features of the viewer.

The HTML parameters and Java Script methods that can be used to customize how annotations work are shown in the *IBM Daeja ViewONE Parameters Reference Manual* and *IBM Daeja ViewONE JavaScript Reference Manual*. You can download these documents from the following web page:

<http://www.ibm.com/support/docview.wss?uid=swg27041437>

Pre-requisite Requirements

The setting up and configuration of the IBM Daeja ViewONE Annotations Module requires you to have already installed and deployed the viewer (either Applet viewer or Virtual viewer) and are able to view images in the viewer.

And unless you are using an existing Enterprise Content Manager system where the annotations interface has already been written and integrated, there is also a requirement for a development resource to write and be responsible for the [server components](#) that interface between the viewer and back-end repository where image and annotation data is stored.

Annotation data is kept separate from image files

IBM Daeja ViewONE separates annotation data from the image files. This has several significant advantages when dealing with annotating documents in a web environment:

- By keeping the annotations separate from image files and in text form, the download and upload of annotations can also be kept separate.
- This in turn means quicker repeat download and upload of annotations because the amount of data is much less than if it was attached to the image file. It also means a true web-based annotation editing system can be deployed because there are no longer large delays receiving and sending image to and from the server.
- It also means it is possible for annotations to be held in a database, and therefore searched upon independently of image files.

Server Component

The implementation of annotations requires both a client-side component and a server-side component.

The client-side component is IBM Daeja ViewONE. The server-side component is required for saving and retrieving the annotation data and should be written and provided by whoever integrates the viewer with the backend system.

This manual provides the necessary specifications for writing of this component.

This document assumes you already know how to install and use the applet without annotations. If this is the first time you have installed IBM Daeja ViewONE, read the relevant setup guide to set up the core viewer before embarking on deploying the

Annotations and "Flow Layout" format documents

Beginning with version 4.0.2 of the viewer, it is possible to view emails and Microsoft Office documents by using either the Universal Viewing module or the Office module. Where you might have viewed a document using one of these modules (for example, the Universal Viewing module) and then view it again using a different module (for example, the email module), you might see a difference in the way the document is laid out and paginated because these modules format layouts differently.

The reason for this is that formats such as email or Word are not explicitly defined formats (such as TIF or PDF) but rather a "flow layout" type format. The nature of these formats means that the layout and pagination of the document itself depends on what is used to view it.

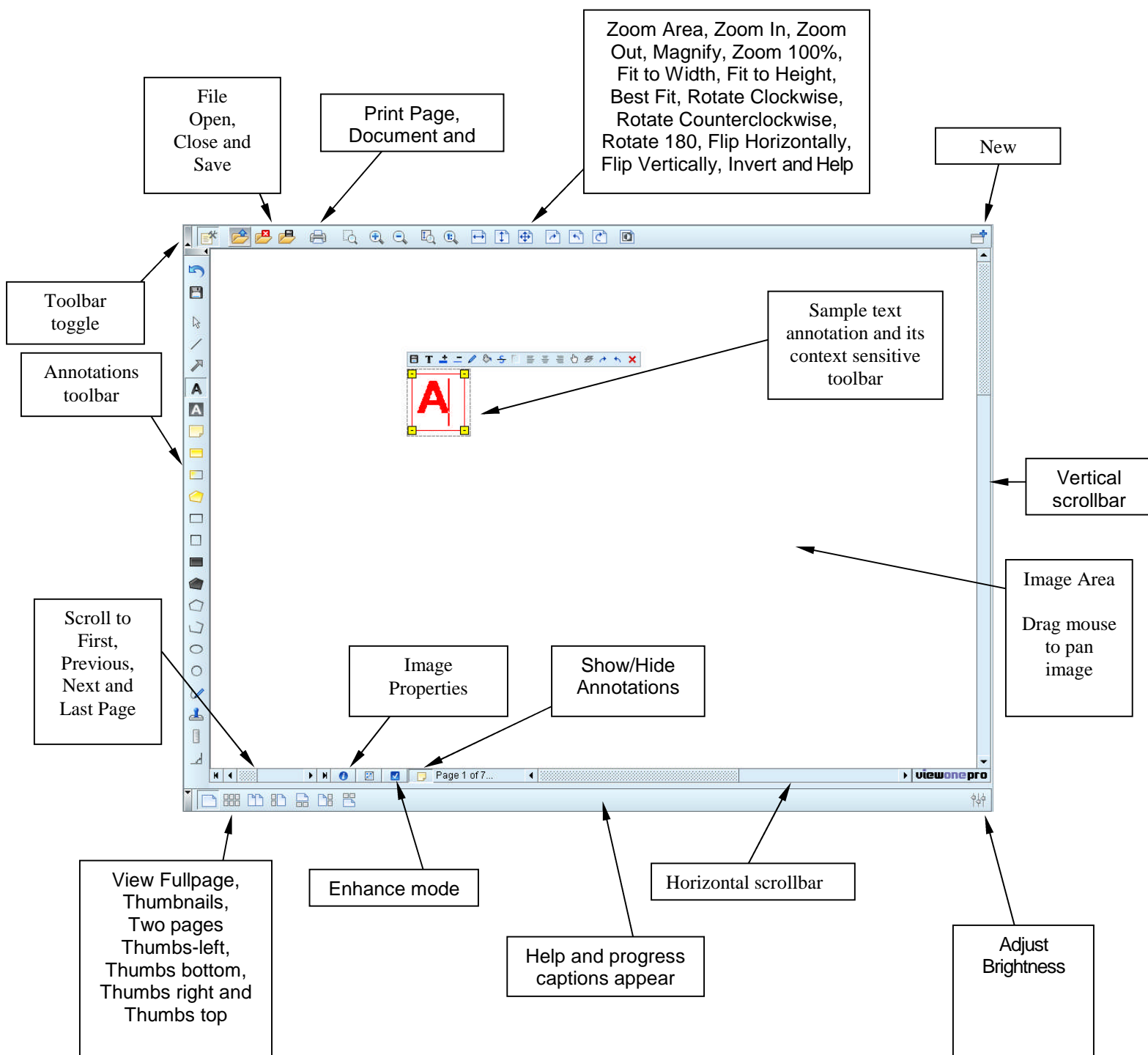
This presents further potential for annotation data to "move" in terms of its location on the document which might represent a serious security risk for customers who are using annotations to securely redact documents.

For example, if you load an email in Microsoft Outlook, the layout of the email itself is altered as soon as you start to adjust the size of the window that contains it – but of course this does not occur if you open a PDF document.

*** IMPORTANT NOTE ***

Given these facts, you should ensure as much as possible that the same module and version are used to view such documents because Daeja cannot guarantee that annotations added to "flow layout" documents will not move at some point in the future further to possible bug fixes and enhancements that might affect the manner in which a document is laid out. The [ANNOTINFO marker](#) introduced in version 4.0.2 of the viewer goes some way to address this by including the version and module used to display the document itself in the annotation data. And as noted above, this is a particularly important consideration when it comes to applying annotations as secure redactions on these document types.

IBM Daeja ViewONE and the Annotations User Interface



Deployment

This section assumes that you have already downloaded and installed the viewer itself – if you have not done so, you should refer to the relevant setup guide to see how to download and deploy the viewer itself.

You can check that the IBM Daeja ViewONE Annotations Module is present by clicking the IBM Daeja ViewONE logo in the viewer and verifying that the module is included in the list of extensions.

After a valid configuration file (lic.v1) with the IBM Daeja ViewONE Annotations Module included has been deployed, there is no installation required. However, you must activate the IBM Daeja ViewONE Annotations Module by including the following parameters in your HTML source code:

<param name="annotate" value="true">

(allows annotations to be viewed but not added or edited)

AND

<param name="annotateEdit" value="true">

(allows annotations to be viewed, added and edited)

These parameters are described in detail in the *IBM Daeja ViewONE Parameters Reference Manual*.

Configuring the Annotations Server Object

This section of the manual describes the server-side object that is required to send and receive annotations in IBM Daeja ViewONE. The object is called when an annotation save operation is carried out by IBM Daeja ViewONE to receive the annotation data that a user might have created or edited.

Whoever implements the system will need to write this server object to interface between the viewer and the backend repository. The object can be written in any server-side object style (such as CGI, ASP, EXE, or servlet), as it only needs the ability to receive and parse an annotations text stream.

With this approach, you can adapt the handling of definitions to meet implementation-specific requirements. For example, the definitions might be written to a database, they might be stored as text files next to their respective images, or they might be passed on again to another object.

If you want to control the permissions of users to view annotation data then this can be achieved through use of a simple or complex security model in conjunction with the appropriate code changes to the server-side object that handles the annotation data. Refer to [Appendix D](#) on Annotation Security for further details on this feature.

The approaches that can be used to implement the annotations server-side save object are described over the next few pages.

Servlet Approach (recommended method)

This approach uses the HTTP Post protocol to deliver the annotation data in one single stream of data.

The location of the server-side object is specified using the “annotationSaveServlet” parameter (described in the *IBM Daeja ViewONE Parameters Reference Manual*). The value of this parameter must be in the form of a URL (relative to the codebase or an absolute value).

Parameters can be added to the query string part of the URL, such as user ID or document ID, to provide user or document-specific annotations, but it is up to the object’s author to process these.

As with any servlet, this needs to be placed on a valid web application server such as JBoss, WebSphere, or WebLogic. Whoever carries out the integration must be familiar with deploying the servlet to your preferred web application server.

At save time, the annotations data is URL encoded and streamed to the servlet.

Note that unlike the chunked POST approach described on the following page, use of the "annotationSaveServlet" parameter means that only the annotation data is sent in a single stream of data.

HTTP:POST Approach

This approach uses the HTTP Post protocol to deliver the annotation data in "chunked" data segments. Therefore, this approach is inherently more complex than the "annotationSaveServlet" approach.

The location of the POST object is specified using the "annotationSavePost" parameter (described in the *IBM Daeja ViewONE Parameters Reference Manual*). The value of this parameter must be in the form of a URL (relative to the codebase or an absolute value).

Parameters can be added to the POST data, such as user ID or document ID to send user or document-specific annotations, by using the "annotationPostPrefix" parameter. IBM Daeja ViewONE then tags on its own annotations data.

At save time, the annotations data is URL encoded and put into a series of segments. Each segment can contain up to 64KB of encoded data. The number of segments generated depends on the amount of annotations data to be sent.

The following parameters are then added to the URL specified by the "annotationSavePost" parameter:

<i>parameter</i>	<i>Description</i>
size	Size in bytes of the URL encoded annotations data being sent.
numdata	Total number of data segments that are to be sent.
data<N>	URL encoded annotations data, where <N> is the sequential identifier for the segment.

The call to the server-side object is then made and the data passed with a content-type of "application/x-www-form-urlencoded".

On the server-side, to retrieve the annotations data from the URL the data<N> segments are concatenated. URL encoded data is normally handled by the web server, so this should not be a concern. The data retrieved can then be checked by comparing the value of the size parameter passed in the URL against the size of the data actually received.

Each time the object is called by IBM Daeja ViewONE the object should send back a confirmation in the form of <OK> for success or <FAILED> for an annotation save failure of some kind.

Customizing the Annotations Module

After you have deployed the IBM Daeja ViewONE Annotations Module, the next step is to make use of the extensive set of HTML parameters and Java Script methods described in the *IBM Daeja ViewONE Parameters Reference Manual* and *IBM Daeja ViewONE JavaScript Reference Manual*.

These parameters and methods allow you to configure the annotations to work in exactly the way you need and to customize the “look and feel” to fit in with the Enterprise Content Manager system you are integrating it into.

[Appendix D](#) describes a specific customizable feature of the annotations module which allows you to configure the annotations security.

Annotations File Format

This section of the manual describes the structure of the text file that is used to define annotations. This is the file that must be supplied to IBM Daeja ViewONE for viewing annotations and is the file format sent to the server by IBM Daeja ViewONE.

General Structure

The annotations file is an ASCII text file with the optional extension of “.ant”. Annotations are defined through a set of properties, some mandatory, some optional, each one defined on a separate line.

Typically, the file is case insensitive except for where text properties are defined. Blank lines are ignored.

The format of this file is similar to a Windows INI file.

The start of an annotation definition is denoted by a left square bracket. This is then followed by a “type” identifier that indicates what kind of annotation is being defined as shown in the following example. The identifier is terminated by a right square bracket. A carriage-return, a linefeed, or a carriage-return and linefeed then delimit the end of the line.

Properties that are specific to the type of annotation defined follow. Only one property can be defined per line.

For example:

```
[ARROW]
X1 = 535
Y1 = 277
X2 = 89
Y2 = 138
PAGE = 1
EDIT = 1
COLOR = 255, 0, 0
```

The end of an annotation definition is denoted by the start of another annotation definition (identified by the '[' and ']' brackets) or by the end of the file.

EMPTY Marker

If no annotations are present in the document when annotations are saved, IBM Daeja ViewONE returns a file with just [EMPTY] inside. When an annotations definition file is loaded it must contain either annotation definitions or the [EMPTY] marker. If it does not, then an error message is displayed and the document is not shown.

ANNOTINFO Marker

Annotations saved in version 4.0.2 or later of the viewer contain the ANNOTINFO annotation and do not load in earlier versions of the viewer. This is because earlier versions of the viewer do not recognize the ANNOTINFO annotation type. If you want to maintain compatibility with an earlier version, for example, for evaluating a new version of the viewer, you can use the `disableAnnotInfo` parameter that is described elsewhere in this manual.

From V4.0.2 of the viewer onwards, whenever a user saves annotation data, by default, it also includes an additional layer of information that is presented in the annotation data stream entitled ANNOTINFO. This section contains detail on the module AND version of that module that was used to view the document when the annotation data was saved. This means that going forward, the viewer can be configured to use the same module (and version) with which the annotation data was added.

However, there are still two issues to consider here:

- 1) Existing annotation data created before 4.0.2 of the viewer does not have the ANNOTINFO marker so the viewer has no knowledge of which module was used to annotate the documents originally. This can mean that annotation data moves if users use a different module to that used to annotate the document originally. For example, the user has viewed and annotated emails by using the Universal Viewing module but has now switched to using the Office module to view these same annotated documents.
- 2) Future enhancements or bug fixes to these modules might cause changes to the way documents are laid out (for example, a fix to an issue with a table in a border on a Word document means that a single page Word document is now a two page Word document).

Depending on what has gone before and also potentially with future changes, in certain situations it is unavoidable that annotations are going to move and potentially even leave the page (described as "out of bounds") if they are viewed with different modules or even different versions of the same module.

To address these potential issues, a number of parameters have been included starting with version 4.0.2 of the viewer to allow tight control of the viewer's behavior when it encounters various scenarios and these are described.

Note that these parameters only affect Office and email documents that have annotations. Other documents always load in the appropriate module detected by the viewer.

Standard mandatory properties

Standard mandatory properties are now described. These properties are required for all annotation types.

page

PAGE = n

Specifies a page for the annotation.

If the value specified is -1 then the annotation is displayed across all pages in the document. To achieve this IBM Daeja ViewONE produces a copy of the annotation for each page of the document. If annotations are then saved, these copies are returned as individual annotations in the definitions file (along with their appropriate page number) and are treated as independent annotations thereafter.

If the value specified is -2 then the behavior is the same except that the annotations are not be treated as independent annotations. A change to the annotation on one of the pages is reflected across all the pages automatically.

The LABEL properties for these new annotations are a copy of the original LABEL property with a page number appended. This makes these annotations unsuitable for use with hyperlinks, as their names change automatically upon creation.

edit

EDIT = 1

Boolean for whether the annotation can be edited. False=0, True=1.

Standard optional properties

Standard optional properties are now described. These properties are optional for any of the annotation types described above except where specifically excluded.

lineWidth

LINEWIDTH = *n*

Specifies the width of the annotation's lines in image pixels.

color

COLOR = *color*

The annotation's color specified in either comma-delimited RGB values or the IBM Daeja ViewONE color scheme text, as listed in [Appendix A](#). The default values are 255, 0, 0 (red), but the default color can be changed by using the “annotationsDefaultLineColor” or “annotationsDefaultTextColor” parameters. See the *IBM Daeja ViewONE Parameters Reference Manual* for details.

label

LABEL = *any text*

Specifies an ID for the annotation. Carriage-returns and linefeeds are not permitted within the value. The default value is the annotation type; for example, “Arrow” or “Oval.” Leading and trailing white space is removed upon save and load operations.

tooltip

TOOLTIP = *any text*

Specifies a tooltip for the annotation. Carriage-returns and linefeeds are not permitted within the value. The value is case-sensitive. There is no default value.

Note: Tooltips are particularly useful when used with annotation hyperlinks.

createDate

CREATEDATE = *dd mm yyyy, hh:mm:ss[, loc±hh:mm]*

For example:

CREATEDATE = 12 Jun 2001, 18:27:05

CREATEDATE = 12 Jun 2001, 18:27:05, GMT+01:00

Specifies the date that the annotation was created. The optional “, *loc±hh:mm*” is a local time zone stamp. “*loc*” stands for location, and is a three letter identifier for the time zone. The “*hh:mm*” specifies the amount the local time was offset from that location’s time zone. The “+” or “-” then specify whether that time is ahead or behind the time zone.

modifiedDate

MODIFIEDDATE = *dd mm yyyy, hh:mm:ss[, loc±hh:mm]*

For example:

MODIFIEDDATE = 12 Jun 2001, 18:27:05

MODIFIEDDATE = 12 Jun 2001, 18:27:05, GMT+01:00

Specifies the date the annotation was last modified. The optional “, *loc±hh:mm*” is a local time zone stamp. “*loc*” stands for location, and is a three letter identifier for the time zone. The “*hh:mm*” specifies the amount the local time was offset from that location’s time zone. The “+” or “-” then specify whether that time is ahead or behind the time zone.

view

VIEW = *n*

Specifies an integer value for the print and view properties of an annotation. The possible values are as follows:

- | | |
|---|---|
| 1 | = viewing only (no printing) |
| 2 | = printing only (no viewing or editing by the user) |
| 3 | = printing and viewing |

The default value is 3.

IBM Daeja ViewONE does not return the property in the definition file if the value is 3.

pageSize

PAGESIZE = *n, n*

Specifies the main page's width and height associated with the annotation. The value breaks down into width and height in that order (from left to right).

It is only required if annotations are to be displayed in thumbnails where thumbnails are specified as separate files (using the “thumb<*N*>” HTML parameter – see the *IBM Daeja ViewONE Parameters Reference Manual*). In such cases, the width and height values must be those for the original main image (not the scaled thumbnail image). IBM Daeja ViewONE is then able to rescale any annotations that might be present on that page to suite the thumbnail image.

pageURL

PAGEURL = *pageURL*

Shows the URL IBM Daeja ViewONE used to retrieve the page the annotation was placed on.

The property is written out by IBM Daeja ViewONE, but is ignored upon reading. It allows server-side processes that handle the annotations definition file to see which image file each annotation (and each page) is associated with.

createdId

CREATEDID = *userID*

Specifies the user ID value that was in use when the annotation was created. The “userID” parameter is described in the *IBM Daeja ViewONE Parameters Reference Manual*.

If the user ID is not set at the time the annotation is saved the property is not returned in the definitions file.

modifiedId

MODIFIEDID = *userID*

Specifies the user ID value that was in use when the annotation was last modified. The “userID” parameter is described in the *IBM Daeja ViewONE Parameters Reference Manual*.

If the user ID is not set at the time the annotation is saved the property is not returned in the definitions file.

blankOutImage

BLANKOUTIMAGE = 1

Boolean for whether the page the annotation is on should be blanked out rather than displayed. It is a security option, allowing an image to be blocked from view, but for its annotations to still be visible.

For example, if a user cannot view a particular page of a document then a simple text annotation might be set up on that page which says “Sorry, viewing denied” and has a BLANKOUTIMAGE value of 1.

The user is not able to edit or modify this annotation, even if the EDIT property is set to 1. Do not include this property unless the value is 1.

customProperty

CUSTOMPROPERTY = *any text*

Specifies custom text to be kept with the annotation. It is not interpreted by IBM Daeja ViewONE, but if it is defined, as IBM Daeja ViewONE reads the annotations file in, it makes a note of the entry and writes it out again at save time.

Just as the CUSTOM annotation type allows information to be associated with an annotation definition file, the CUSTOMPROPERTY allows information to be associated with individual annotations.

Note: The property is only written back at save time if it came from a valid annotation definition.

hyperlink

HYPERLINK = <annotation type><hyperlink property>[<hyperlink property>]

An annotation hyperlink is an action that is taken when an annotation is “clicked” on by the user (or double-clicked – if the “annotationDbClick” HTML parameter is used, or if IBM Daeja ViewONE has the annotations toolbar open).

This action can be any of four types: to change to a specific page, to go to a specific annotation, to call a JavaScript function or to open a web page. Each of these types has additional parameters, described in the following pages.

We have tried to make setting up and using annotation hyperlinks as simple as possible (the easiest way to set them up is to use the IBM Daeja ViewONE user interface - see context-sensitive toolbar for annotations). The following pages describe the format of the hyperlink properties.

Before trying to define your own hyperlinks in a definitions file it is worth having a play with the IBM Daeja ViewONE user interface to see how they work. You might find that, for example, a JavaScript hyperlink is more suitable for some tasks than the web hyperlink. Both can be used to open a web page (for example), but using a JavaScript hyperlink would mean more flexibility over what that window looks like.

JavaScript hyperlinks can also be used for any other task, such as opening another document, or some custom job. JavaScript hyperlinks however, require the use of the IBM Daeja ViewONE JavaScript event-handler (see the *IBM Daeja ViewONE JavaScript Reference Manual*) and so are a little more complex to initially set up.

Using the IBM Daeja ViewONE annotation tooltips to help with hyperlinks

Any annotation can have a tooltip.

A tooltip is some text that appears when the user ‘hovers’ the mouse over an item (in this case an annotation). If you have a hyperlink defined for that annotation, for example, to jump to another page in the document, then it would be helpful to the user if a tooltip was also defined for that annotation. Perhaps to inform the user of what happens when the annotation is clicked, for example, “Click here to see page 5”, or “Click here to see the index.”

The four types of hyperlink are as follows:

Page hyperlinks	Hyperlinks between pages in a document (internal document hyperlinks).
Annotation hyperlinks	Hyperlinks between annotations in a document (internal document hyperlinks).
JavaScript hyperlinks	Hyperlink to call the JavaScript event handler.
Web page hyperlinks	Hyperlink to another web page (external hyperlink).

Page hyperlinks

HYPERLINK = <page><17>

Page hyperlinks require the text <page> followed by the <page number>. With all hyperlink properties the “<” and “>” must be included as shown. Pages outside of the document’s page range are ignored.

Annotation hyperlinks

HYPERLINK = <annotation><highlight1>

Annotation hyperlinks require the text <annotation> followed by the <annotation label> of the other annotation that it is to be linked to.

If a link is made to a label name that is shared by two or more annotations within a document, IBM Daeja ViewONE links to the first of those annotations.

When this hyperlink is used, IBM Daeja ViewONE scrolls the page to the upper left of the linked annotation (or as close as it can get), while using the current scale preference (fit to height, fit to width, and so on.).

However, if the linked annotation has any window settings associated with it, then they are used instead, for example, a particular zoom factor, scroll position, or rotation angle. See description of the HYPERLINKSETTINGS property.

JavaScript hyperlinks

HYPERLINK = <javascript><help text>

JavaScript hyperlinks require the text <javascript> followed by <any text>. When the annotation is clicked IBM Daeja ViewONE calls the designated JavaScript event handler (see the *IBM Daeja ViewONE JavaScript Reference Manual* on how to define a JavaScript event handler). The <any text> parameter is used as the event handler's "text" property, and can be interpreted in any way you choose (e.g., as a document id, web page, comment, etc.). The event id is 23.

This hyperlink is useful for designing implementation-specific hyperlink behavior. For example, upon receiving the hyperlink's text string, the JavaScript event handler might display a dialog box with help information, tell the server something, or even use IBM Daeja ViewONE JavaScript calls to open a new document (see *IBM Daeja ViewONE JavaScript Reference Manual* for further detail).

Web page hyperlinks

HYPERLINK = <web><http://www.mysite.com/page.html><_self>

Web page hyperlinks require the text <web> followed by the <web page url> then the <target> for the new web page. The target can be any of the four default values assigned by IBM Daeja ViewONE (see [Appendix C](#)) or a custom target, as defined by the "annotationTarget" HTML parameter (see *IBM Daeja ViewONE Parameters Reference Manual*).

The target allows the choice of which "window" displays the new web page; <_self> for example, causes the browser to change the current web page containing IBM Daeja ViewONE, whereas <_blank> opens a new window containing the new web page.

The web page URL can be relative to the IBM Daeja ViewONE codebase parameter or it can be absolute. If the HTML parameter "annotationHyperlinkWeb" is used, then when a relative URL is used it is relative to the base address as defined by that HTML parameter.

hyperlinkSettings

HYPERLINKSETTINGS = <settingsValue>

This property defines “window” settings for an annotation:

If this property is included with an annotation and that annotation is linked to by another (by using the hyperlink property of the other annotation to link to this annotation), then the window settings defined by this property are used on activation of that hyperlink.

The format of this property is quite lengthy, but it allows a wide range of window settings to be defined. If you are unsure what value to use for this property, the easiest way to get a value is to use the IBM Daeja ViewONE user interface.

Select an annotation that you want to link to. Zoom in, scroll, rotate, etc. so that the page appears in the way you want it to be viewed.

Then use the right mouse button to click the annotation (to select it in edit mode), and click the “H” hyperlink button of the context toolbar that appears.

Click the “Grab window settings” button, save the annotation. You need to note the annotations “label” so that you know which annotation to link to when you define the hyperlink from another annotation.

Now look at the annotations definition file that IBM Daeja ViewONE has created (this might be on the server if you use the “annotationSave” HTML parameter, or in the “ANT” file if the image was loaded from a local file).

Locate the annotation and note the HYPERLINKS value included for the window settings that you used.

The format for this property is as follows:

view#scale#flip#rotation#invert#zoom#scrollx#scrolly#brightness#contrast#luminance

Where:

<i>view</i>	Currently ignored but reflects the view mode used (use a value of 1)
<i>scale</i>	0 (fit to width) or 1 (fit to height) or 2 (best fit) or 3 (zoom)
<i>flip</i>	0 (no flip) or 1 = flip horizontally or 2 = flip vertically or 3 (flip both ways)
<i>rotation</i>	0 or 90 or 180 or 270 (angle in degrees)
<i>invert</i>	0 (not color inverted) or 1 (colors inverted)
<i>zoom</i>	Zoom factor (for example, 2.0)
<i>scrollX</i>	Percentage of image to scroll in the x-axis
<i>scrollY</i>	Percentage of image to scroll in the y-axis
<i>brightness</i>	0 to 512 (where 0 =dark, 512 = light and 256 is the middle typical value)
<i>contrast</i>	0 to 512 (256 is the middle typical value)
<i>luminance</i>	0 to 512 (256 is the middle typical value)

Example value: 1#3#0#90#1#2.0#20#14#256#256#512

Annotation types and their properties

Annotation-specific properties are now described. For each annotation type there are both mandatory and optional properties.

The standard optional properties described in the previous section apply to all annotation types except where specifically excluded (for example, LINECOLOR).

Some default values for both specific and standard properties can be changed using HTML parameters or Java Script methods described in the relevant reference manual.

Description: Arrow

Defines an arrow annotation. There are two points in the definition. The arrow head is drawn at the first point.

Definition: [ARROW]

(Mandatory)

$X1 = n$	X coordinate of the arrow's first point. n is an image pixel value on the image's X axis.
$Y1 = n$	Y coordinate of the arrow's first point. n is an image pixel value on the image's Y axis.
$X2 = n$	X coordinate of the arrow's second point.
$Y2 = n$	Y coordinate of the arrow's second point.

(Optional)

$ARROWHEADSIZE = n$	Specifies the arrow's head size. n has a minimum value of 1 and no fixed upper value. The default value is 1.
---------------------	---

See standard optional properties:

The minimum LINEWIDTH for this annotation is 1.

Example: [ARROW]
X1 = 537
Y1 = 277
X2 = 89
Y2 = 138
PAGE = 1
EDIT = 1
ARROWHEADSIZE = 1

Description:**Custom**

Defines a custom annotation type. The custom annotation type is not really an annotation, but is a way to add extra information to an annotations definition file.

This annotations type cannot be added from the user interface, but can be generated through a server-side process that adds instances to a definitions file. An instance is not interpreted by IBM Daeja ViewONE, but if one is defined, as IBM Daeja ViewONE reads the annotations file in, it makes a note of the entry and writes it out again at save time (though it is not certain the entry will be in the same place when written out again).

The CUSTOM type allows server-side processes to keep track of custom information between definition time and save time, without IBM Daeja ViewONE interpreting it in the meantime.

Definition:

[CUSTOM]

(Mandatory)

There are no mandatory properties.

(Optional)

There are no optional properties that apply.

Example:

[CUSTOM]

This text shows that you can put anything into a CUSTOM annotation.

Text 1

Text 2

Text 3

I was generated by server "nautilus1" @ 18:31 GMT on 27/03/02

Description:**Freehand**

Defines a freehand line. There must be a minimum of three points in the definition.

Definition:

[FREEHAND]

(Mandatory)

$X1 = n$	X coordinate of the line's first point. n is an image pixel value on the image's X axis.
$Y1 = n$	Y coordinate of the line's first point. n is an image pixel value on the image's Y axis.
$X2 = n$	X coordinate of the line's second point.
$Y2 = n$	Y coordinate of the line's second point.
$X3 = n$	X coordinate of the line's third point.
$Y3 = n$	Y coordinate of the line's third point.

(Optional)

$Xn = n$	X coordinate of point n on the line.
$Yn = n$	Y coordinate of point n on the line.

See standard optional properties:

The minimum LINEWIDTH for this annotation is 1.

Example:

[FREEHAND]

X1 = 408

Y1 = 940

X2 = 408

Y2 = 940

X3 = 412

Y3 = 936

X4 = 412

Y4 = 927

X5 = 425

Y5 = 907

PAGE = 1

EDIT = 1

Description: **Highlight Rectangle**

Defines a rectangular highlight.

Definition: [HIGHLIGHT]

(Mandatory)

$X = n$	X coordinate of the rectangle's upper left corner. n is an image pixel value on the image's X axis.
$Y = n$	Y coordinate of the rectangle's upper left corner. n is an image pixel value on the image's Y axis.
$WIDTH = n$	Width of the rectangle in image pixels.
$HEIGHT = n$	Height of the rectangle in image pixels.

(Optional)

$LINEWIDTH = 0$	The LINEWIDTH property is ignored if it is not set to 0.
$FILLCOLOR = color$	The highlight color specified in either comma delimited RGB values or the IBM Daeja ViewONE color scheme text, as listed in Appendix A . If no color is defined then the highlight is invisible.
$TRANSPARENT = 0$	Boolean for whether the highlight is transparent. False=0, True=1. The default value is 0.
$ASPECTRATIO=1:1$	When set to 1:1, forces the highlight to be a square

See standard optional properties:

The minimum LINEWIDTH for this annotation is 0 (default).

Example: [HIGHLIGHT]
X = 674
Y = 61
WIDTH = 310
HEIGHT = 176
PAGE = 1
EDIT = 1
LINEWIDTH = 0
FILLCOLOR = 255, 255, 255
TRANSPARENT = 0

Description: **Highlight Polygon**

Defines a highlight polygon. There must be a minimum of three points in the definition.

Definition: [HIGHLIGHTPOLYGON]

(Mandatory)

$X1 = n$	X coordinate of the polygon's first point. n is an image pixel value on the image's X axis.
$Y1 = n$	Y coordinate of the polygon's first point. n is an image pixel value on the image's Y axis.
$X2 = n$	X coordinate of the polygon's second point.
$Y2 = n$	Y coordinate of the polygon's second point.
$X3 = n$	X coordinate of the polygon's third point.
$Y3 = n$	Y coordinate of the polygon's third point.

(Optional)

$Xn = n$	X coordinate of point n in the polygon.
$Yn = n$	Y coordinate of point n in the polygon.
$LINEWIDTH = 1$	The $LINEWIDTH$ property is ignored if it is not set to 1.
$FILLCOLOR = color$	The highlight color specified in either comma delimited RGB values or the IBM Daeja ViewONE color scheme text, as listed in Appendix A. If no color is defined then the highlight is invisible.
$TRANSPARENT = 0$	Boolean for whether the highlight is transparent. False=0, True=1. The default value is 0.

See standard optional properties:

The minimum $LINEWIDTH$ for this annotation is 0 (default 1).

Example: [HIGHLIGHTPOLYGON]

$X1 = 1246$
 $Y1 = 61$
 $X2 = 1581$
 $Y2 = 220$
 $X3 = 1356$
 $Y3 = 371$
 $X4 = 1144$
 $Y4 = 200$
 $PAGE = 1$
 $EDIT = 1$
 $LINEWIDTH = 1$
 $FILLCOLOR = 255, 255, 255$
 $TRANSPARENT = 0$

Description: **Line**

Defines a line. There are two points in the definition.

Definition: [LINE]

(Mandatory)

$X1 = n$	X coordinate of the line's first point. n is an image pixel value on the image's X axis.
$Y1 = n$	Y coordinate of the line's first point. n is an image pixel value on the image's Y axis.
$X2 = n$	X coordinate of the line's second point.
$Y2 = n$	Y coordinate of the line's second point.

(Optional)

See standard optional properties:

The minimum LINEWIDTH for this annotation is 1.

Example:

[LINE]
X1 = 498
Y1 = 167
X2 = 85
Y2 = 65
PAGE = 1
EDIT = 1

Description:**Note**

Defines a note and its text.

Definition:

[NOTE]

(Mandatory)

$X = n$	X coordinate of the rectangle's upper left corner. n is an image pixel value on the image's X axis.
$Y = n$	Y coordinate of the polygon's first point. n is an image pixel value on the image's Y axis.
$WIDTH = n$	Width of the rectangle in image pixels.
$HEIGHT = n$	Height of the rectangle in image pixels.
$TEXT = \text{any text}$	Specifies text for the note. New lines are indicated by the <N> identifier. Leading and trailing white space is removed upon save and load operations.

(Optional)

$FILLCOLOR = \text{color}$	The note's color (background color) specified in either comma delimited RGB values or the IBM Daeja ViewONE color scheme text, as listed in Appendix A . The default values are 255, 255, 153 (pale yellow).
$TRANSPARENT = 0$	Boolean for whether the note is transparent. False=0, True=1. The default value is 0.
$RECTANGULAR = 0$	Boolean for whether the note style is rectangular (that is, without leaf turn up). The default is 0.

See standard optional properties: LINEWIDTH and HYPERLINK not available.

Example:

[NOTE]

$X = 81$

$Y = 323$

$WIDTH = 50$

$HEIGHT = 65$

$TEXT = \text{This is line 1}<N>\text{This is line 2}$

$PAGE = 1$

$EDIT = 1$

$FILLCOLOR = 255, 255, 255$

$TRANSPARENT = 0$

$RECTANGULAR = 0$

Description:**Open Polygon**

Defines an open-ended polygon. There must be a minimum of three points in the definition.

Definition:

[OPENPOLYGON]

(Mandatory)

$X1 = n$	X coordinate of the polygon's first point. n is an image pixel value on the image's X axis.
$Y1 = n$	Y coordinate of the polygon's first point. n is an image pixel value on the image's Y axis.
$X2 = n$	X coordinate of the polygon's second point.
$Y2 = n$	Y coordinate of the polygon's second point.
$X3 = n$	X coordinate of the polygon's third point.
$Y3 = n$	Y coordinate of the polygon's third point.

(Optional)

$Xn = n$	X coordinate of point n in the polygon.
$Yn = n$	Y coordinate of point n in the polygon.

See standard optional properties:

The minimum LINEWIDTH for this annotation is 1.

Example:

[OPENPOLYGON]

X1 = 1242

Y1 = 670

X2 = 1246

Y2 = 878

X3 = 927

Y3 = 882

X4 = 1017

Y4 = 658

PAGE = 1

EDIT = 1

Description: **Oval**

Defines an oval.

Definition: [OVAL]

(Mandatory)

$X = n$	X coordinate of the oval's center. n is an image pixel value on the image's X axis.
$Y = n$	Y coordinate of the oval's center. n is an image pixel value on the image's Y axis.
$WIDTH = n$	The oval's horizontal radius in image pixels.
$HEIGHT = n$	The oval's vertical radius in image pixels.

(Optional)

$FILLCOLOR = color$	The oval's fill color specified in either comma delimited RGB values or the IBM Daeja ViewONE color scheme text, as listed in Appendix A . The default values are 255, 255, 255 (white).
$TRANSPARENT = 0$	Boolean for whether the oval is transparent. False=0, True=1. The default value is 1.
$ASPECTRATIO=1:1$	When set to 1:1 forces the highlight to be a circle (V3-Only)

See standard optional properties:

The minimum LINEWIDTH for this annotation is 0.

Example:

[OVAL]
X = 257
Y = 752
WIDTH = 143
HEIGHT = 130
PAGE = 1
EDIT = 1
FILLCOLOR = 255, 255, 255
TRANSPARENT = 0

Description:**Polygon**

Defines a closed polygon. There must be a minimum of three points in the definition.

Definition:

[POLYGON]

(Mandatory)

$X1 = n$	X coordinate of the polygon's first point. n is an image pixel value on the image's X axis.
$Y1 = n$	Y coordinate of the polygon's first point. n is an image pixel value on the image's Y axis.
$X2 = n$	X coordinate of the polygon's second point.
$Y2 = n$	Y coordinate of the polygon's second point.
$X3 = n$	X coordinate of the polygon's third point.
$Y3 = n$	Y coordinate of the polygon's third point.

(Optional)

$Xn = n$	X coordinate of point n in the polygon.
$Yn = n$	Y coordinate of point n in the polygon.
$FILLCOLOR = color$	The polygon's fill color specified in either comma delimited RGB values or the IBM Daeja ViewONE color scheme text, as listed in Appendix A . The default values are 255, 255, 255 (white).
$TRANSPARENT = 0$	Boolean for whether the polygon is transparent. False=0, True=1. The default value is 1.

See standard optional properties:

The minimum LINEWIDTH for this annotation is 0.

Example:

[POLYGON]

$X1 = 1528$

$Y1 = 457$

$X2 = 1634$

$Y2 = 694$

$X3 = 1442$

$Y3 = 809$

$X4 = 1360$

$Y4 = 621$

PAGE = 1

EDIT = 1

FILLCOLOR = 255, 255, 255

TRANSPARENT = 0

Description: **Rectangle**

Defines a rectangle.

Definition: [RECTANGLE]

(Mandatory)

$X = n$	X coordinate of the rectangle's upper left corner. n is an image pixel value on the image's X axis.
$Y = n$	Y coordinate of the rectangle's upper left corner. n is an image pixel value on the image's Y axis.
$WIDTH = n$	Width of the rectangle in image pixels.
$HEIGHT = n$	Height of the rectangle in image pixels.

(Optional)

$FILLCOLOR = color$	The rectangle's fill color specified in either comma delimited RGB values or the IBM Daeja ViewONE color scheme text, as listed in Appendix A . The default values are 255, 255, 255 (white).
$TRANSPARENT = 0$	Boolean for whether the rectangle is transparent. False=0, True=1. The default value is 1.
$ASPECTRATIO=1:1$	When set to 1:1 forces the highlight to be a square (V3-only)

See standard optional properties:

The minimum LINEWIDTH for this annotation is 0.

Example:

[RECTANGLE]
X = 690
Y = 269
WIDTH = 282
HEIGHT = 209
PAGE = 1
EDIT = 1
FILLCOLOR = 255, 255, 255
TRANSPARENT = 0

Description: **Redaction**

Defines a rectangular redaction (a filled rectangle).

Definition: [REDACT]

(Mandatory)

$X = n$	X coordinate of the rectangle's upper left corner. n is an image pixel value on the image's X axis.
$Y = n$	Y coordinate of the rectangle's upper left corner. n is an image pixel value on the image's Y axis.
$WIDTH = n$	Width of the rectangle in image pixels.
$HEIGHT = n$	Height of the rectangle in image pixels.

(Optional)

$LINEWIDTH = 0$	The LINEWIDTH property is ignored if it is not set to 0.
$FILLCOLOR = color$	The redaction's color specified in either comma delimited RGB values or the IBM Daeja ViewONE color scheme text, as listed in Appendix A . The default values are 0, 0, 0 (black).
$TRANSPARENT = 0$	Boolean for whether the redaction is transparent. False=0, True=1. The default value is 0.
$ASPECTRATIO=1:1$	When set to 1:1 forces the highlight to be a square

See standard optional properties:

The minimum LINEWIDTH for this annotation is 0 (default).

Example: [REDACT]
X = 1058
Y = 371
WIDTH = 225
HEIGHT = 221
PAGE = 1
EDIT = 1
LINEWIDTH = 0
FILLCOLOR = 255, 255, 255
TRANSPARENT = 0

Description:**Redaction Polygon**

Defines a redaction polygon (a filled polygon). There must be a minimum of three points in the definition.

Definition:

[REDACTPOLYGON]

(Mandatory)

$X1 = n$	X coordinate of the polygon's first point. n is an image pixel value on the image's X axis.
$Y1 = n$	Y coordinate of the polygon's first point. n is an image pixel value on the image's Y axis.
$X2 = n$	X coordinate of the polygon's second point.
$Y2 = n$	Y coordinate of the polygon's second point.
$X3 = n$	X coordinate of the polygon's third point.
$Y3 = n$	Y coordinate of the polygon's third point.

(Optional)

$Xn = n$	X coordinate of point n in the polygon.
$Yn = n$	Y coordinate of point n in the polygon.
$LINEWIDTH = 1$	The $LINEWIDTH$ property is ignored if it is not set to 1.
$FILLCOLOR = color$	The redaction's color specified in either comma delimited RGB values or the IBM Daeja ViewONE color scheme text, as listed in Appendix A . The default values are 0, 0, 0 (black).
$TRANSPARENT = 0$	Boolean for whether the redaction is transparent. False=0, True=1. The default value is 0.

See standard optional properties:

The minimum $LINEWIDTH$ for this annotation is 0 (default 1).

Example:

[REDACTPOLYGON]

$X1 = 702$
 $Y1 = 519$
 $X2 = 948$
 $Y2 = 621$
 $X3 = 845$
 $Y3 = 792$
 $X4 = 645$
 $Y4 = 756$
 $PAGE = 1$
 $EDIT = 1$
 $LINEWIDTH = 1$
 $FILLCOLOR = 255, 255, 255$
 $TRANSPARENT = 0$

Description:**Stamp**

Defines an image overlay (an overlaid image which might be in any of the image formats supported by IBM Daeja ViewONE).

Definition:

[STAMP]

(Mandatory)

$X = n$	X coordinate of the rectangle's upper left corner. n is an image pixel value on the image's X axis.
$Y = n$	Y coordinate of the rectangle's upper left corner. n is an image pixel value on the image's Y axis.
RESOURCE = image: <i>image file</i>	Path (absolute or relative) to the stamp's image file. The path must have the "image:" prefix. Spaces are permitted in the path.

(Optional)

SCALE = n	Specifies the scale multiplier. 100% = 1.0, 200% = 2.0, and so on. The default value is 1.0.
RESOURCE = image:mytif.tif< <i>color</i> >	A comma delimited RGB value or IBM Daeja ViewONE color scheme text can be added to the end of the RESOURCE property to specify the transparent color for the stamp. The values need to be contained within the < and > delimiters. This color is then treated as transparent and the background image is visible through the areas where this color occurs. This can be a convenient way to overlay transparent images, such as logos or watermarks.
ROTATION = n	The stamp's rotation specified in degrees. Valid values for n are 0, 90, 180, 270, 360. The default is 0.
FONTHEIGHT = n	Specifies the font height to use for the stamp text. n is an image pixel value. The minimum acceptable value is 1.

See standard optional properties: COLOR and LINEWIDTH not available.

Example:

```
[STAMP]
X = 134
Y = 1017
SCALE = 1.0
RESOURCE = image:mytif.tif<255, 255, 255>
ROTATION = 180
PAGE = 1
EDIT = 1
LINEWIDTH = 0
```

Description:**Text**

Defines a text overlay.

Text annotations are displayed using a predefined font. There are several fonts, each based on Arial, of different sizes. It is important to note that IBM Daeja ViewONE does not rely on the font configuration of the user's machine. The font is loaded from a font resource file specific to IBM Daeja ViewONE that is supplied as part of the IBM Daeja ViewONE installation.

Font size is specified through the FONTHEIGHT property.

Two properties are provided to allow for rotation of text annotations so that the most appropriate one can be used when integrating into existing systems. TEXTROTATION is the recommended property.

Definition:

[TEXT]

(Mandatory)

$X = n$	X coordinate of the rectangle's upper left corner. n is an image pixel value on the image's X axis.
$Y = n$	Y coordinate of the rectangle's upper left corner. n is an image pixel value on the image's Y axis.
TEXT= <i>any text</i>	Specifies text for the overlay. New lines are indicated by the <N> identifier. Leading and trailing white space is removed upon save and load operations.
FONTHEIGHT = n	Specifies the font height to use for the overlay's text. n is an image pixel value. The minimum acceptable value is 1.

(Optional)

FILLCOLOR = <i>color</i>	<p>The overlay's fill color (background color) specified in either comma delimited RGB values or the IBM Daeja ViewONE color scheme text, as listed in Appendix A. This property only comes into effect where the TRANSPARENT property is set to 0. The default values are 255, 255, 153 (pale yellow).</p> <p>If the TRANSPARENT property is set to 1 and the PAGE specified is a color image, the text is given a visible fill color using the values 255, 255, 255 (white), as color transparencies are not supported.</p>
ROTATION = n	The text's rotation specified in degrees. Valid values for n are 0, 90, 180, 270, 360. This differs from the TEXTROTATION property in that the rotation is clockwise relative to the page. The default is 0.
TEXTROTATION = n	The text's rotation specified in degrees. A valid value is an integer between 0 and 359. This differs from the ROTATION property in that the rotation is counterclockwise relative to the page. The default is 0.

TRANSPARENT = 0	Boolean for whether the overlay is transparent. False=0, True=1. The default value is 1.
SEMITRANSSPARENT = 0	Boolean for whether the text is semi-transparent. False=0, True=1. The default value is 0. If the text is set to semi-transparent then the color used for the overlay's fill color is also forced to semi-transparent.
STRIKETHROUGH = 0	Specifies if the text is struck through, that is, the text has a line through the center of it. (IBM Daeja ViewONE Standard V3.0.332 or later, IBM Daeja ViewONE Professional V1.0.332 or later)

See standard optional properties: LINEWIDTH not available.

Example:

```
[TEXT]
X = 85
Y = 265
TEXT = This is line 1<N>This is line 2
FONTHEIGHT = 34
PAGE = 1
EDIT = 1
FILLCOLOR = 255, 255, 0
ROTATION = 0
TEXTROTATION = 90
TRANSPARENT = 0
SEMITRANSSPARENT = 0
STRIKETHROUGH = 1
```

Appendix A

IBM Daeja ViewONE Color Scheme Text

Where parameters require a color value, it is possible to use either a comma delimited RGB value or the IBM Daeja ViewONE color scheme text. The text values that can be used in place of RGB values within those parameters are listed below. The values are case insensitive.

<i>Text</i>	<i>Corresponding RGB values</i>
Black	0, 0, 0
Blue	0, 0, 255
Cyan	0, 255, 255
DarkGray	64, 64, 64
Gray	128, 128, 128
Green	0, 255, 0
LightGray	192, 192, 192
Magenta	255, 0, 255
Orange	255, 200, 0
Pink	255, 175, 175
Red	255, 0, 0
White	255, 255, 255
Yellow	255, 255, 0

Appendix B

IBM Daeja ViewONE Default Stamps

The IBM Daeja ViewONE default stamp values and their properties are as follows:

<i>annotationStamp parameter value</i>	<i>annotationStampProperties parameter value</i>
<date>	<menu=Today's date>
RECEIVED	<menu=Received>
PAID	<menu=Paid>
APPROVED	<menu=Approved>
* REJECTED *	<menu=Rejected>
* VOID *	<menu=Void>

Appendix C

IBM Daeja ViewONE Default Web Hyperlink Targets

The IBM Daeja ViewONE default web hyperlink target values are as follows:

<i><target><menu label text></i>
<i><_self><Same window></i>
<i><_blank><New window></i>
<i><_parent><Parent frame></i>
<i><_top><Top frame></i>

Appendix D

Annotation Security

This section of the manual describes the two methods that are available for implementing annotation security. Integrators are free to choose whichever method is best suited to their security requirements. This section explains how the security model works and the integrator will need to code the annotation server-side component(s) to handle the additional security data in conjunction with the logged on user ID so that the permissions are handles correctly.

Additional HTML parameters that can be used to configure annotation security are described in the *IBM Daeja ViewONE Parameters Reference Manual*.

Simple Annotation Security

Simple annotation security uses the “edit” and “view” annotation properties to define whether a user can edit, view or print a particular annotation.

The “edit” annotation property is not included, by default. Therefore, when a user creates and saves an annotation by using the user-interface, the 'annotations definitions file' contains details about the annotation that do not include the "edit" property because that user has 'edit' privileges for that annotation. When another user retrieves the annotation file without the "edit" property being specifically included, that user can edit that annotation.

However, the server object that serves up the annotation file might insert the 'edit' property for any annotation. Setting this property to 0 (zero) prevents that user from editing that particular annotation. This can be done selectively for each annotation.

To make use of this functionality, you must have a server object (such as an exe, an asp, or a JSP) that allows you to control the content of the annotation file when it is served out to users.

There is also an HTML parameter option "annotateEdit" (described in the *IBM Daeja ViewONE Parameters Reference Manual*) that can be used to disable the edit option for all annotations. This is a global option and when used, the user concerned is not able to edit or modify any annotation.

The same principle can be applied for the “view” annotation property. The default value is “3”, allowing all users to both view and print the annotation. Annotations created by the user have this default value and when saved the “view” property is not set in the annotations definition file.

The server object can then set the view property to “1” to allow viewing of the annotation but not printing or “2” to allow printing but not viewing.

Extended Annotation Security

The simple annotation security options are limited in that users do not have any control over the privileges that are assigned to annotations they create. Also, there is no way to individually control modification, deletion, and reading privileges separately.

It is possible to enable 'extended' options that provide both additional user options and the ability to specify more detailed privileges.

The extended option relies on you using the "userId" HTML parameter (see the *IBM Daeja ViewONE Parameters Reference Manual*), then adding an HTML parameter called "annotationSecurityModel" (see the *IBM Daeja ViewONE Parameters Reference Manual*) with a value of "2".

Example HTML:

```
<param name="userId" value="user1">  
<param name="annotationSecurityModel" value="2">
```

This enables, as default, extended security options for each new annotation that is created by the user.

When the user saves annotations, extra properties are written to the annotations file, as follows:

```
SECURITYMODEL = 2  
READ = 1  
MODIFY = 1  
EXECUTE = 1  
PRINT = 1  
DELETE = 1  
PASSWORDMODIFY = *****  
PASSWORDSECURITY = *****  
OWNER = Whoever  
MODIFYSECURITY = 1
```

Note that there are separate privileges for read, modify, execute, print, delete, and modify-security. Each of these can be changed by either the user or the server object (when serving the file back to the user).

By default, they are all set to 1 (enabled). The "MODIFYSECURITY" property indicates whether the user has the ability to edit these privileges using the user interface. If enabled, when the user selects the annotation, an additional 'padlock' context button appears. If users click this button and they see a 'security' dialog box that they can use to change any of the preceding security properties.

The owner of an annotation is the user that created the annotation in the first place, identified by the "owner" property also written to the annotations file. If this owner property matches the "userId" HTML parameter, then that user always has access to this security dialog.

Alternatively, the HTML parameter "userAdmin" (see the *IBM Daeja ViewONE Parameters Reference Manual*) can be specified. If set to "true", then irrespective of the "owner" and "userId" values, that user has 'admin' privileges, and is able to edit annotation security for any annotation.

If a user creates an annotation, then pulls up this dialog box and clears the 'modify' option, no user (except themselves and admin users) are able to modify that annotation (for example, they cannot select it or move it). If modify is left selected, but the 'delete' option is cleared, then other users can modify the annotation but they cannot delete it. If the 'read' option is disabled, the other users do not get to see the annotation. If the 'print' option is disabled then users are not able to print the annotation.

The 'execute' option is used for annotation 'hyperlinks.' That is, if an annotation has a hyperlink, but 'execute' is disabled, then other users are not able to use that hyperlink.

If a document has existing annotations that were created using the 'simple' method, then the extended options can be enabled on them by adding the "SECURITYMODEL = 2" line to each annotation (at retrieval time) by the server object serving the annotations file.

If the "annotationEditPasswordModify" parameter is set to "true" (see the *IBM Daeja ViewONE Parameters Reference Manual*), then it is possible to set a password that must be entered before a user can modify an annotation. If the "annotationEditPasswordSecurity" parameter is set to "true" (see the *IBM Daeja ViewONE Parameters Reference Manual*), then it is possible to set a password that must be entered before a user can modify annotation security.

The passwords are saved in the annotations file, using the properties PASSWORDMODIFY and PASSWORDSECURITY. They are private-key 32-bit encrypted so they are not able to viewable or editable except through IBM Daeja ViewONE. If a user forgets a password assigned to an annotation, if they are the OWNER or an ADMIN user then they can clear and re-enter a password. All other users cannot, unless they enter the correct modify annotation security password.

The passwords are not displayed anywhere (the "*" character is used during entry and on the new dialog).

Notices

This information was developed for products and services offered in the U.S.A.

IBM® may not offer the products, services, or features discussed in this document in other countries. Consult your local IBM representative for information on the products and services currently available in your area. Any reference to an IBM product, program, or service is not intended to state or imply that only that IBM product, program, or service may be used. Any functionally equivalent product, program, or service that does not infringe any IBM intellectual property right may be used instead. However, it is the user's responsibility to evaluate and verify the operation of any non-IBM product, program, or service.

IBM may have patents or pending patent applications covering subject matter described in this document. The furnishing of this document does not grant you any license to these patents. You can send license inquiries, in writing, to:

IBM Director of Licensing
IBM Corporation
North Castle Drive Armonk, NY 10504-1785
U.S.A.

For license inquiries regarding double-byte (DBCS) information, contact the IBM Intellectual Property Department in your country or send inquiries, in writing, to:

Intellectual Property Licensing
Legal and Intellectual Property Law
IBM Japan Ltd.
19-21, Nihonbashi-Hakozakicho, Chuo-ku
Tokyo 103-8510, Japan

The following paragraph does not apply to the United Kingdom or any other country where such provisions are inconsistent with local law:

INTERNATIONAL BUSINESS MACHINES CORPORATION PROVIDES THIS PUBLICATION "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESS OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF NON-INFRINGEMENT, MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE. Some states do not allow disclaimer of express or implied warranties in certain transactions, therefore, this statement may not apply to you.

This information could include technical inaccuracies or typographical errors. Changes are periodically made to the information herein; these changes will be incorporated in new editions of the publication. IBM may make improvements and/or changes in the product(s) and/or the program(s) described in this publication at any time without notice.

Any references in this information to non-IBM websites are provided for convenience only and do not in any manner serve as an endorsement of those websites. The materials at those websites are not part of the materials for this IBM product and use of those websites is at your own risk.

IBM may use or distribute any of the information you supply in any way it believes appropriate without incurring any obligation to you.

Licensees of this program who wish to have information about it for the purpose of enabling: (i) the exchange of information between independently created programs and other programs (including this one) and (ii) the mutual use of the information which has been exchanged, should contact:

IBM Corporation
J46A/G4
555 Bailey Avenue
San Jose, CA 95141-1003
U.S.A.

Such information may be available, subject to appropriate terms and conditions, including in some cases, payment of a fee.

The licensed program described in this document and all licensed material available for it are provided by IBM under terms of the IBM Customer Agreement, IBM International Program License Agreement or any equivalent agreement between us.

This information contains examples of data and reports used in daily business operations. To illustrate them as completely as possible, the examples include the names of individuals, companies, brands, and products. All of these names are fictitious and any similarity to the names and addresses used by an actual business enterprise is entirely coincidental.

Any performance data contained herein was determined in a controlled environment. Therefore, the results obtained in other operating environments may vary significantly. Some measurements may have been made on development-level systems and there is no guarantee that these measurements will be the same on generally available systems. Furthermore, some measurements may have been estimated through extrapolation. Actual results may vary. Users of this document should verify the applicable data for their specific environment.

Information concerning non-IBM products was obtained from the suppliers of those products, their published announcements or other publicly available sources. IBM has not tested those products and cannot confirm the accuracy of performance, compatibility or any other claims related to non-IBM products. Questions on the capabilities of non-IBM products should be addressed to the suppliers of those products.

All statements regarding IBM's future direction or intent are subject to change or withdrawal without notice, and represent goals and objectives only.

This information contains examples of data and reports used in daily business operations. To illustrate them as completely as possible, the examples include the names of individuals, companies, brands, and products. All of these names are fictitious and any similarity to the names and addresses used by an actual business enterprise is entirely coincidental.

COPYRIGHT LICENSE:

This information contains sample application programs in source language, which illustrate programming techniques on various operating platforms. You may copy, modify, and distribute these sample programs in any form without payment to IBM, for the purposes of developing, using, marketing or distributing application programs conforming to the application programming interface for the operating platform for which the sample programs are written. These examples have not been thoroughly tested under all conditions. IBM, therefore, cannot guarantee or imply reliability, serviceability, or function of these programs.

Trademarks

IBM, the IBM logo, and ibm.com are trademarks or registered trademarks of International Business Machines Corp., registered in many jurisdictions worldwide. Other product and service names might be trademarks of IBM or other companies. A current list of IBM trademarks is available on the Web at "Copyright and trademark information" at <http://www.ibm.com/legal/us/en/copytrade.shtml>.

Adobe, the Adobe logo, PostScript, and the PostScript logo are either registered trademarks or trademarks of Adobe Systems Incorporated in the United States, and/or other countries.

Microsoft, Windows, and Windows NT are trademarks of Microsoft Corporation in the United States, other countries, or both.

Java and all Java-based trademarks and logos are trademarks or registered trademarks of Oracle and/or its affiliates.

Other company, product, and service names may be trademarks or service marks of others.

Privacy policy considerations

IBM Software products, including software as a service solutions, ("Software Offerings") may use cookies or other technologies to collect product usage information, to help improve the end user experience, to tailor interactions with the end user or for other purposes. In many cases no personally identifiable information is collected by the Software Offerings. Some of our Software Offerings can help enable you to collect personally identifiable information. If this Software Offering uses cookies to collect personally identifiable information, specific information about this offering's use of cookies is set forth below.

Depending upon the configurations deployed, this Software Offering may use cookies that collect each user's user name for purposes of session management, authentication, and enhanced user usability. These cookies cannot be disabled.

If the configurations deployed for this Software Offering provide you as customer the ability to collect personally identifiable information from end users via cookies and other technologies, you should seek your own legal advice about any laws applicable to such data collection, including any requirements for notice and consent.

For more information about the use of various technologies, including cookies, for these purposes, See IBM's Privacy Policy at <http://www.ibm.com/privacy> and IBM's Online Privacy Statement at <http://www.ibm.com/privacy/details> the section entitled "Cookies, Web Beacons and Other Technologies" and the "IBM Software Products and Software-as-a-Service Privacy Statement" at <http://www.ibm.com/software/info/product-privacy>.



Product Number: 5725-Q02, 5725-Q03, 5725-Q04

SC27-6392-00

