

XIOview V1.5

Image Zoom Applet – TRIAL VERSION

Copyright © 2005 by Martin Rieger, XIO® Software. XIO® is a registered trademark of Martin Rieger. Java™ is a registered trademark of Sun Microsystems, Inc.. All other trademarks are properties of their respective owners.

Documentation

(1.0 - 12/27/2005)

Visit www.xio.biz for the most actual information on *XIOview* and other XIO® Software products.

By using the *XIOview* applet you indicate that you have read, understood, and agreed to the DISTRIBUTION AND USAGE AGREEMENT (see the last page of this document). - If you have questions regarding licensing, please do not hesitate to contact XIO® Software.

This TRIAL VERSION allows testing the applet on local computers. For testing *XIOview* on a website prior to purchasing a license, you can request a 30-day-activation-key. Please contact XIO® Software via the support section at www.xio.biz and specify the domains on which you want to test the applet (up to five).

TABLE OF CONTENTS

INTRODUCTION / FUNCTIONALITY OVERVIEW	2
CHANGES SINCE VERSION 1.0 / PROGRAM FILES	3
GENERAL HTML IMPLEMENTATION	4
PARAMETERS	
General Usage	5
Changes since Version 1.0	6
Parameter Overview	6
JAVASCRIPT - CONTROL	12
PERFORMANCE ISSUES	13
PROBLEMS & TROUBLESHOOTING	14
SUPPORT & CONTACT	15
DISTRIBUTION AND USAGE AGREEMENT	16

INTRODUCTION

Thank you for using *XIOview 1.5*!

In order to fully describe all (optional) features of the applet this documentation has grown quite large. However, configuring XIOview is generally not difficult nor very time consuming. In many cases it is not necessary to read the entire document.

- **If you have some experience in implementing applets you can also directly test and modify the examples delivered with this release.**
- **Users of the previous version (*XIOview V1.0*) should take a look at page 3 and 6 to get informed of important changes.**

I believe that in its current version, *XIOview* is the best, easiest to use, and most functional applet of its kind available today. After searching the web and comparing it to the few available other "zoom applets", I have found none of comparable quality and ease of use.

Therefore I am really confident that *XIOview 1.5* will prove useful for your applications.

Sincerely,

Martin Rieger, XIO® Software

FUNCTIONALITY OVERVIEW

XIOview is a Java-applet with zoom functionality and various other features for embedding and interactive viewing of image content on websites. It is especially useful for presenting high-resolution images on small viewing space by providing a scaled overview of the entire image and at the same time allowing user-guided instant exploration of all details through zooming.

Additionally, *XIOview* provides image filtering capabilities, allowing display-adjustment of the source image to different viewing environments. Numerous of (optional) parameters can be modified – within the HTML-document or by using the realtime JavaScript-Interface – making *XIOview* extremely adaptable to almost any media solution.

Apart from this *XIOview* has a very easy & intuitive user-interface and is simple to implement.

- Zoom into any area of an image (JPG or GIF) with just one mouseclick / scroll through the image by moving the mouse
- Adjust zoom-level instantly using arrow keys (arrow-up = zoom in / arrow-down = zoom out)
- Image-filters: brightness, contrast, RGB-channel, gamma, grayscale, inversion
- Adaptable: zoom size, background-color, selection-marker, (smooth) scaling...

CHANGES SINCE VERSION 1.0

From version 1.0 to 1.5, *XIOview* underwent massive development, especially concerning internal code. The following list shortly outlines some of the most important changes:

- entirely reprogrammed using JDK 1.1.8 (Version 1.0 was programmed using JDK 1.0.2)
- processing, performance and exception-handling improved
- because of few usage, external (non-program) synchrony-problems in various testing environments and increased configuration complexity **the optional *XIOviewControl* applet was removed** (the existing JavaScript-Interface fully substitutes all of its options)
- added new functionality, such as realtime keyboard zoom-control; link on right click; brightness, contrast and gamma correction; color inversion
- improved functionality of RGB-filtering (now percentage-steps for each channel from 0-200%)
- progress bar for loading status of the image
- all of the applet's code is now packaged into a ".jar" and a ".cab"-file (the ".cab"-file is provided as support for older versions of MS Internet Explorer only) – no additional ".class"-files
- a license-key for each acquired domain-specific license ("usage right") is used for verification

For better consistency some of the parameter names and structures have been changed, others are still supported, but marked as deprecated. - See the "Parameters"-Section (Subsection on page 6: "*Changes since Version 1.0*") of this document.

PROGRAM FILES

The program code of *XIOview 1.5* is provided in two different compressed archive file formats:

- *XIOview.jar* - the official Java-archive format (compressed with the "*jar*"-Tool of JDK 1.1.8)
- *XIOview.cab* - the cabinet-archive format from Microsoft (using "*cabarc*" from Microsoft)

Simply copy both files in the same directory from where they are to be executed. The following section ("*General HTML-Implementation*") explains how make a correct reference to the program's code in a HTML-document. If you already have experience implementing Java-applets in HTML you may skip this section.

- The file "*readme.txt*" contains a short overview of all important files delivered with this release. -

GENERAL HTML IMPLEMENTATION

In this section the basic integration of the *XIOview* applet into a HTML-document is illustrated. If you already know how to implement applets in HTML you may proceed directly to the "Parameters"- section (page 5).

The implementation example shown here uses the old HTML 3.2 `<applet>`-tag, which relies on the browser's built in Java Virtual Machine (JVM) to execute the code. Until today this standard seems to be the most reliable and compatible.

It is also possible to directly reference the Java plug-in by using the `<object>`-tag. But since *XIOview* is programmed in Java 1.1.x (supported by almost any modern Java-enabled browser) it is generally not necessary to download or use a plugin of a higher (more actual) Java version.

The following code sample shows a valid implementation of the applet, displaying the most important attributes and the essentially required parameters:

```
<applet code="biz.xio.XIOview" archive="XIOview.jar" codebase="applet/" name="applet1"
width="320" height="240" alt="XIOview 1.5 Java Applet (requires a Java-enabled browser)">

  <param name="cabbase" value="XIOview.cab">

  <param name="license_key" value="your-domain-specific-key">
  <param name="image" value="images/your-image.jpg">

  <!-- more optional parameters here -->

  <!-- The text below is shown if Java is not currently supported/activated. -->
  <p>
    The interactive content of this page requires a Java-enabled
    browser.<br>
    Please check your browser's settings and activate Java.<br>
    If your browser has no built in Java plugin you can obtain one at
    <a href="http://www.java.com/en/download/">www.java.com</a>.
  </p>
</applet>
```

Attributes within the <applet>-tag:

- `code="biz.xio.XIOview"` refers to the main class of the applet which will be executed.
- `archive="XIOview.jar"` refers to the compressed archive file ("*jar*" = *J*ava *a*rchive) which contains all executable *.class*-files of the applet (i.e. the entire applet binaries).
- `codebase="applet/"` refers to the location of the applet's code (here it is the file *XIOview.jar*). In this case the applet's code is stored in the subdirectory "*applet/*", relative to the implementing HTML-document. **Note:** In many cases you can also only use the "*archive*"-tag (e.g. *archive = "applet/XIOview.jar"*). But if you use the "*codebase*"-tag, your image file (see parameter "*image*") must be located in the same or in a lower directory as the "*codebase*"-directory.
- `name="applet1"` represents the logical name of the applet, it is used for JavaScript-access. By using this tag (with different name-values) it is possible to implement more than one applet in a HTML-document and controlling each one independently.
- `width="320"` and `height="240"` define the width and height of the applet in pixels.
- `alt="XIOview 1.5 ..."` defines an alternative text if the applet is not executed by the browser.

Essentially required parameters:

For a complete and detailed description of all parameters refer to the "Parameters"-section.

- `<param name="cabbase" value="XIOview.cab">` is a *MS Internet Explorer*-specific parameter which refers to a different compressed archive file format ("cab" = cabinet file), because the official *.jar*-format is not supported by older versions of *MS Internet Explorer*.
- `<param name="license_key" value="your-domain-specific-key">` is a required parameter for all domain-specific licenses of *XIOview*. For every domain *XIOview* was licensed to, you will receive a corresponding code-key-value. This allows launching the applet on this domain.
- `<param name="image" value="images/your-image.jpg">` specifies an image to load (any JPG or GIF still image). Without this parameter correctly set, an error will occur.

PARAMETERS

General Usage:

Every parameter is coded inside the `<applet>`-tags according to the following scheme:

```
<param name="parameter_name" value="this parameter value">
```

Please ensure that all parameters are coded correctly according to this scheme.

If the name of a parameter is incorrect the parameter will not be found by the applet and its value cannot be initialized.

In case the value of a parameter is wrong or invalid *XIOview* will send an exception-message to the standard output (for a Java-enabled browser this generally is the "Java Console"). For optional parameters, *XIOview* will try to use a corrected or default value to allow launching nevertheless.¹²

So if a setting does not work or not work as expected, open the Java Console to check for exception-messages and verify the coding of parameters and their corresponding values.

(1) The required parameter *cabbase* is not interpreted by the applet itself. This parameter is interpreted by MS Internet Explorer (MSIE) only. Since older versions of MSIE do only support ".cab"-files as a compressed archive file format (which contains the applet's code), it is highly recommended – though not literally required – to use this parameter. Consequently there will be no exception messages by the applet if the parameter is wrongly implemented – the applet's code simply will not be found by the concerned browsers and therefore the applet will not launch.

(2) The value of the required parameter *license_key* is only relevant if *XIOview* is launched online (i.e. from a website). On a local computer the applet will launch regardless of the value.

All parameters - except *cabbase* and *license_key* - can also be instantly modified using JavaScript. See the JavaScript-section for usage instructions.

Changes since Version 1.0

This subsection provides a short overview for users of the previous *XIOview* version regarding parameter syntax. Although many changes were made for better name consistency, most basic settings of *XIOview V1.0* will still work with *V1.5*.

The following parameters are **new** in version 1.5:

"*license_key*"¹, "*zoom_smooth*", "*zoom_range*"², "*zoom_permanent*", "*zoom_keyboard_control*", "*zoom_step_percentage*", "*image_brightness*", "*zoom_brightness*", "*image_contrast*", "*zoom_contrast*", "*image_gamma*", "*zoom_gamma*", "*image_inverted*", "*zoom_inverted*", "*on_right_click*".

- (1) A **required** parameter. (Required for using the *XIOview* on a website.)
- (2) Was originally part of *XIOviewControl V1.0*.

The following parameters are **no longer supported** in version 1.5:

"*zoom_up_smooth*"

"*zoom_down_smooth*" – both replaced by "*zoom_smooth*"

"*img_smooth*" – the image will now always be smooth-scaled

The following parameters have been **renamed** and marked as deprecated, but are still supported in version 1.5:

"*img*" to "*image*"

"*selector*" to "*selection_marker*"

"*zoom_width*" to "*zoom_area_width*"

"*show_filename*" to "*filename_show*"

"*zoom_height*" to "*zoom_area_height*"

"*bgcolor*" to "*background_color*"

"*img_rgb*" to "*image_rgb*"

"*autozoom*"* to "*zoom_permanent*"

"*img_grayscale*" to "*image_grayscale*"

(*) *XIOview V1.0b* only

The following parameters have a **changed or extended value format**:

("*img_rgb*") / "*image_rgb*" – now requires three numerical percentage values

"*zoom_rgb*" – now requires three numerical percentage values

("*zoom_width*") / "*zoom_area_width*" – optional support for percentage values

("*zoom_height*") / "*zoom_area_height*" – optional support for percentage values

Parameter Overview:

All parameter names are formatted **bold**. Alternative parameter names are separated by "/". Parameters which are deprecated, though still supported in *XIOview 1.5* for backward-compatibility reasons are formatted small. These names should no longer be used for new implementations, because they may be unsupported in future versions of *XIOview*.

Not implemented optional parameters are automatically initialized with the value(s) indicated in the "preset"-field. - So you need to implement only those whose values you want to change.

1. Required Parameters

The following three parameters must be correctly set to allow functioning of the *XIOview* applet.

name:	cabbase
value:	The cabinet archive file (for MSIE), which contains the applet's code: " <i>XIOview.cab</i> ".
Refers to the additionally provided compressed archive file <i>XIOview.cab</i> (instead of <i>XIOview.jar</i>), required by older versions of <i>MS Internet Explorer</i> (without support for the standardized ".jar"-format). This parameter is interpreted by <i>MS Internet Explorer</i> only.	

name:	license_key
value:	A domain-specific code key, e.g.: " <i>Wd7yXgQU392wGrTL1Snz+64qqvA6xm1HTgWF...</i> ".
This parameter sets the domain-specific license key of <i>XIOview</i> . Without this parameter correctly set the applet will not launch on your website, though it will still work on your local computer ("offline-mode"). NOTE: The key must be entered as an unmodified continuous string, i.e. without any whitespaces or line breaks . Otherwise the license key will be rejected.	

name:	image / img
value:	A JPG or GIF still image file to be loaded, e.g.: " <i>images/myImage.gif</i> ".
Specifies the source image file to be displayed by <i>XIOview</i> . All paths are relative to the HTML-document in which the applet is implemented. It is recommended to use only standard ASCII-characters and to avoid extremely long paths or filenames, which may cause system-dependent referencing problems.	

2.1. Optional Parameters: Zoom Settings

name:	zoom		
value:	One percentage-value within the range specified by the " <i>zoom_range</i> "-parameter (default range setting: 50%-800%), e.g.: " <i>125%</i> ".	preset:	"100%"
Sets the zoom level as percentage of the original source image size. This parameter can be additionally modified through the arrow-keys if the " <i>zoom_keyboard_control</i> "-parameter is set " <i>true</i> " (the default setting). Acceptable values for <i>zoom</i> depend on the " <i>zoom_range</i> "-parameter (default range setting: 50%-800%). NOTE: the actually "perceived zoom effect" also depends on the size of the applet itself: In any case the applet's size is smaller than the source image there will be an zoom effect for <i>zoom</i> -values of 100% or even (much) less, because the source image is automatically scaled to fit the applet's dimensions. So the "perceived zoom effect" equals: $([\text{source image size}] / [\text{applet size}]) * [\text{zoom}]$			

name:	zoom_range		
value:	Two percentage-values, separated by "-", e.g.: "100%-400%".	preset:	"50%-800%"
Defines the range of allowed values for the "zoom"-parameter. The maximum range is 1%-5000%. The settings of the "zoom_range"-parameter provide range limitations for both arrow-key and JavaScript-modifications of the "zoom"-parameter (i.e. any attempt to change the zoom-parameter to a value out of the defined zoom_range will be blocked) .			

name:	zoom_smooth		
value:	A combination of <u>two</u> boolean values, each either "true" or "false", e.g.: "true, true".	preset:	"true, false"
Switches smooth scaling mode for zoom. If set "true", the applet will calculate intermediate pixel values for better optical quality (bilinear filtering / area average-filtering). The first value represents the setting for zoom levels >100%. The second for zoom levels <100%. NOTE: This parameter is highly relevant to rendering performance (especially for zoom levels <100%). If the zoom area (parameters "zoom_area_width" and "zoom_area_height") is large, it is recommended to set both parameters to "false".			

names:	zoom_area_width / zoom_width		
	zoom_area_height / zoom_height		
value:	A numeric pixel value (minimum: 10 pixels) <u>or</u> a percentage value (1-100%), e.g.: "150px" or "75%".	preset:	"100px"
These parameters adjust width and height of the zoomed area. You can use either pixel or percentage (of total applet width / height) values. By setting "zoom_area_width" and "zoom_area_height" to 100%, the zoomed area will fill the entire applet.			

name:	zoom_selection / zoom_selector		
value:	Either "true" or "false".	preset:	"true"
Switches the selection-marker on/off. The selection-marker indicates the size of the image section around the mouse-courser that will be zoomed if the mouse is clicked.			

name:	zoom_permanent / autozoom		
value:	Either "true" or "false".	preset:	"false"
Sets activation-mode of the zoom function. If set "true", zooming will stay permanently activated after a mouseclick until the mouse is clicked again (or the cursor has left the applet area). If set "false", zooming is activated as long the mousebutton is hold down.			

name:	zoom_keyboard_control		
value:	Either "true" or "false".	preset:	"true"
Switches control of zoom-level through arrow-keys on/off. If "true", pressing the arrow keys will increase (arrow up) or decrease (arrow down) zoom-level by the amount defined by the parameter "zoom_step_percentage". Limits for the zoom-level are defined by the "zoom_range"-parameter.			

name:	zoom_step_percentage		
value:	One percentage-value, e.g.: "10%".	preset:	"25%"
Sets the amount by which each keypress of the arrow keys will increase (arrow up) or decrease (arrow down) the zoom level. Limits for the zoom-level are defined by the "zoom_range"-parameter.			

2.2 Optional Parameters: Filter Settings for Image Area and Zoom

names:	image_grayscale / image_greyscale / img_grayscale / img_greyscale		
	zoom_grayscale / zoom_greyscale		
value:	Either "true" or "false".	preset:	"false"
Switches grayscale mode of the displayed image and zoomed area.			

names:	image_rgb / img_rgb		
	zoom_rgb		
value:	Set of three percentage-values, separated by comma - each within a range from 0% to 200%, e.g.: "25%, 140%, 100%".	preset:	"100%, 100%, 100%"
Adjusts the intensity of the red, green, and blue color channel of the displayed image and the zoomed area. 100% is the standard (= unmodified) value for each channel. Values below 100% indicate less intensity (0% = the channel is switched off), values above 100% indicate a higher intensity of the specific channel (200% = the intensity of the channel is doubled). If this parameter is implemented, always all three channel values must be set.			

names:	image_brightness / img_brightness		
	zoom_brightness		
value:	One percentage-value within a range of 0% to 200%, e.g.: "110%".	preset:	"100%"
Adjusts the brightness of the displayed image and the zoomed area. 100% is the standard (= unmodified) value. Values below 100% reduce brightness (0% = no brightness, the area is black), values above 100% increase it (200% = doubled original brightness).			

names:	image_contrast / img_contrast		
	zoom_contrast		
value:	Either one or three percentage-value(s) within a range of 0% to 200%, e.g.: "110%" <u>or</u> "10%, 150%, 90%".	preset:	"100%"

Adjusts the contrast of the displayed image and the zoomed area. By setting only one value, contrast is adjusted equally across all color-channels. By setting three (different) values, contrast will be selectively adjusted for each color-channel (red, green and blue) at the specified amount. *100%* is the standard (= unmodified) value. Values below *100%* reduce contrast (*0%* = minimum contrast), values above *100%* increase it (*200%* = maximum contrast).

names:	image_gamma / img_gamma		
	zoom_gamma		
value:	Either one or three floating-point value(s) between 0.1 to 9.9, e.g.: "2.25" or "1.8, 1.5, 0.35".	preset:	"1.0"

Adjusts the gamma-value of the displayed image and the zoomed area. By setting only one value, gamma-correction is performed equally across all color-channels. By setting three (different) values, gamma-correction will be selectively performed for each color-channel (red, green and blue) at the specified amount.

1.0 is the standard (= unmodified) value. Values below *1.0* reduce gamma (*0.1* = minimum), values above *1.0* increase gamma (*9.9* = maximum).

names:	image_inverted / img_inverted		
	zoom_inverted		
value:	Either "true" or "false".	preset:	"false"

Switches inversion-mode of the displayed image and zoomed area.

2.3. Optional Parameters: General Settings

name:	background_color / bgcolor		
value:	Three numerical values, separated by comma, each within a range of 0-255, e.g.: "100, 50, 220"	preset:	"128, 128, 128"
Sets the background-color of the applet in RGB-values (red, green, blue).			

name:	on_right_click		
value:	Two text-values, separated by comma. Either "zoom, x" or a specified file (with absolute or relative URL) combined with a target-modifier, e.g.: "images/my-image.gif, _self" or "http://www.anywebsite.com/document.htm, _blank"	preset:	"zoom, x"

Specifies an action if the user is right-clicking on the applet's area. Predefined is zooming ("zoom, x"). Alternatively, the applet can request the browser to load a document (e.g. an HTML- or an image-file). This can be achieved by setting an URL (as first value) with a target-modifier (as second value).

Documents can be loaded either from relative (i.e. relative to the HTML-document in which the applet is implemented) or absolute URLs.

NOTE: Two values are always required (even though not always used). - **Any absolute URL must contain "http://"**.

The target-modifier indicates in which HTML-frame or browser window the document shall be displayed (see listing below).

target-modifier: *...opens the document:*

"_blank"	-	...in a new, unnamed window
"_self"	-	...in the same frame and window as the applet
"_parent"	-	...in the applet's parent frame (If the applet's frame has no parent frame, the effect is the same as "_self")
"_top"	-	...in the top-level frame of the applet's window (If the applet's frame is the top-level frame, the effect is the same as "_self")
"anyname"	-	...in the window or frame named "anyname". If a window/frame with this name does not exist, a new window with this name will be created.

name:	filename_show / show_filename		
value:	Either "true" or "false".	preset:	"true"
Sets, if the filename of the source image will be displayed while loading.			

JAVASCRIPT - CONTROL

All of the applet's parameters - **except "cabbase" and "license_key"** - can be modified at runtime using JavaScript.

To call the applet's configuration method it is required to give a name to the implemented applet(s) first. This is done by setting the "*name*"-attribute inside the `<applet>`-tag:

```
<applet code="XIOview.class" archive="XIOview.jar" codebase="applet/" name="applet_name"
width="320" height="240" alt="XIOview 1.5 Java Applet (requires a Java-enabled browser)">

<!-- parameters here -->

</applet>
```

Now the "setConfiguration"-method of the *XIOview* applet exemplar can be accessed via JavaScript commands. Correct commands have the form:

```
document.applet_name.setConfiguration('parameter_name = parameter_value');
```

Instead of "setConfiguration" you can also use the shorthand form "setCfg":

```
document.applet_name.setCfg('parameter_name = parameter_value');
```

It is possible to change various parameters at the same time. In this case a '+' is inserted between each assignment:

```
document.applet_name.setConfiguration('parameter_name1 = value1 + parameter_name2 =
value2 + parameter_name3 = value3 + ...');
```

All parameter names and corresponding values resemble those used for the applet's HTML implementation.

Examples

- `document.any_name.setConfiguration('image = another.jpg + background_color = 0,0,0');`

Requests the *XIOview applet* exemplar called "*any_name*" to load "*another.jpg*" and to change the background color to black.

- `load new image`

Implements a text-link. If clicked, the JavaScript-command will request the *XIOview* applet called "*any_name*" to load "*new.gif*".

- ``

This graphical button-link simultaneously sends a series of commands to two different *XIOview* applets on the same page:

First the applet called "*applet1*" is requested to load "*advertising.gif*" and to enable linking to "*http://www.my-advertising.net*" if the right mouse button is pressed inside the applet's area.

Additionally the applet called "*applet2*" is requested to set the zoom-level to *400%*, to set the gamma-correction of the image-area and the zoomed-area to *0.8*, and to set zooming to grayscale mode.

Annotations

All JavaScript parameter modifications made during the loading process of an image will be executed once the image is completely loaded and rendered. - **Except the "*image*"-parameter**, which will stop the current loading process and try to load the newly specified image.

Depending on the used browser the first JavaScript-method call could need a bit more time to execute due to browser-dependent initialization processes. All subsequent calls will then be executed without further delay.

PERFORMANCE ISSUES

Apart from the client's system environment (hardware, browser), rendering performance and memory consumption of *XIOview* mainly depend on the following factors:

1. Source image size. affects: **memory**, performance

Large images require more memory than smaller ones and need more time to be scaled to the applet's dimensions after loading. The relation is exponential: By doubling width and height of the image, memory requirements for image-buffering will be 4 times higher.

2. Zoom-area dimensions. affects: **performance**
parameters: "*zoom_area_width*", "*zoom_area_height*"

Larger zoom-areas require more computing power than smaller ones. The relation is exponential: Doubling the zoom-area dimensions will increase rendering requirements by a factor of 4. Especially in combination with smooth zooming and zoom-filtering (see 3. and 5.), this parameter is highly relevant to performance.

3. <u>Smooth zooming</u> .	<u>affects</u> :	performance
	<u>parameter</u> :	" <i>zoom smooth</i> "

Deactivating smooth zooming may substantially increase rendering speed. Especially for zoom levels <100% (the second of the two boolean parameter values) and large images, rendering requirements can be very high, because a huge number of pixels has to be processed for area average-filtering (the smaller the zoom level, the larger the number of pixels).

4. Applet dimensions. affects: memory, **performance**

If applet dimensions increase, the time for redrawing the applet area will also increase. Memory consumption is also higher for a large-scaled applet.

5. <u>Zoom-filtering</u>	<u>affects:</u>	performance
	<u>parameters:</u>	"zoom_grayscale", "zoom_rgb", "zoom_brightness", "zoom_contrast", "zoom_gamma", "zoom_inverted"

Using realtime zoom filter algorithms requires extra rendering operations. However the resulting increase in rendering requirements is not as pronounced as compared to smooth zooming (3.).

PROBLEMS & TROUBLESHOOTING

If the applet does not launch at all, see if you have set the correct "*code*", "*archive*", and "*codebase*" – attributes inside the `<applet>`-tag. Further check if your browser has an installed and activated *Java Runtime Environment* (JRE).

If the applet freezes at the beginning of image loading the problem most certainly originates from a security exception triggered by an impermissible file access request. Security settings do not permit accessing files on a higher directory-level than the applet's "*codebase*"-directory.

Most problems originate from mistyped or wrongly implemented parameters and values. If a setting does not work it is generally a good idea to check if all parameters are coded correctly. Additionally – if values of a parameter are invalid – *XIOview* will send exception messages to the standard output (which in most cases is the Java-Console). Open the Java-Console to check for these messages.

Generally do not forget to code the right tag and attribute-terminators (e.g. `<`, `>`, `/>`, `"` ...).

Usage-related problems

- The use of many applets on the same page is discouraged, as this may break down system performance and/or the *Java Virtual Machine* may run short of memory (in worst case an

OutOfMemoryException will be triggered and further execution will be stopped).

Compatibility

The applet is designed to launch on any Java-enabled browser whose *Java Runtime Environment* (JRE) or *Java Virtual Machine* (JVM) supports at least the 1.1.x Java version. Compatibility can be expected for the following browsers:

- *MS Internet Explorer* since version 5.x with integrated JVM
- *Mozilla Firefox* with Java-plugin
- *Netscape Navigator* since version 4.05 with integrated JVM
- *Netscape Communicator* since version 6.x with Java-plugin
- *Safari* on OS X
- *iCab* 2.8.2 on OS X
- *Opera* with Java-plugin (in many cases already delivered with the browser)
- *HotJava* since version 1.1

...and others

In fact, *XIOview* will run instantly on the vast majority of client systems without installing anything. In the remaining cases, *Java* support can generally be easily obtained by installing the Java-plugin. The Java-plugin is available freely at <http://www.java.com>.

XIOview V1.5 has proven high stability in various testing environments. However due to the heterogeneity of *Java Runtime Environments* (JRE) on different systems (e.g. *MS Internet Explorer integrated Java Virtual Machine* versus *Sun JRE-Virtual Machine*) minor differences in the applet's performance or behavior may occur. These slight differences are not caused by the applet itself and generally do not impair functioning.

SUPPORT & CONTACT

If you have problems regarding the implementation of *XIOview* that you could not resolve by reading this manual you can request e-mail support.

Please try to give a detailed and accurate description of the problem. This makes helping easier.

Your feedback on *XIOview* is also appreciated. Feel free to share it.

Please refer to the support-section of XIO® Software's website: <http://www.xio.biz>

There you also find the most actual information on *XIOview* and other XIO® Software products.

DISTRIBUTION & USAGE AGREEMENT - TRIAL VERSION

This free trial version ("evaluation version") of the software ("XIOview", "the applet"), its documentation and all other related items are being provided by Martin Rieger ("XIO Software", "the copyright holder") under the following Distribution and Usage Agreement ("license"). By obtaining and using this software, you ("the tester" / "the distributor") agree that you have read, understood, and will comply with the following terms and conditions.

1.1 This evaluation version of the software is provided free of charge. The tester may freely use this evaluation version to test the applet's functions and suitability for the intended area of application without obligation of purchase.

1.2 The evaluation version allows unrestricted testing of all functions of the applet on local computers. To test the applet online (on a website) prior to purchase, a free 30-day-activation-key can be requested by specifying the domain-name(s) (up to 5) on which testing is desired to XIO Software. However, XIO Software is not obligated to grant this request in any case.

2. The tester may also freely redistribute this trial version under the following conditions:

- (a) The software and all its associated parts (documentation, license, images, etc.) must be redistributed complete and unmodified.
- (b) The software and all its associated parts must be delivered free of any charge. (If you want to redistribute this program on a paid Shareware-CD or other media, please first ask XIO Software for permission.)
- (c) The distributor must refer to "XIO Software" as the original copyright holder of the program.

3. It is prohibited to decompile, reverse-engineer or modify any part of the software's code in any way.

4. The tester is not allowed to use this software under any unlawful circumstances (e.g. by displaying unlawful contents), or using the program in a way that may negatively affect the software's or its copyright holder's reputation.

5. The registered trademark "XIO"(R) and the product name "XIOview" may not be used to endorse or promote other software products, or products derived therefrom, except with prior written permission by XIO Software.

6. THIS SOFTWARE AND DOCUMENTATION IS PROVIDED "AS IS", AND THE COPYRIGHT HOLDER MAKES NO REPRESENTATIONS OR WARRANTIES, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO, WARRANTIES OF MERCHANTABILITY OR FITNESS FOR ANY PARTICULAR PURPOSE OR THAT THE USE OF THE SOFTWARE OR DOCUMENTATION WILL NOT INFRINGE ANY THIRD PARTY PATENTS, COPYRIGHTS, TRADEMARKS OR OTHER RIGHTS.

7. THE COPYRIGHT HOLDER WILL NOT BE LIABLE FOR ANY DIRECT, INDIRECT, SPECIAL OR CONSEQUENTIAL DAMAGES ARISING OUT OF ANY USE OF THE SOFTWARE OR ITS ASSOCIATED PARTS.

Copyright (c) 2005 by Martin Rieger, XIO Software. XIO(R) is a registered trademark of Martin Rieger. All other trademarks are properties of their respective owners.