nexImage Actions and Commands

Date	23.01.2007
Document Release	1.7
nexImage Release	2.3.1
Copyright	nexbyte gmbh

Contents

Introduction

This document deals with actions and their respective commands. Actions can be employed in many ways. Actions make it possible to simplify the editing process of repetitive tasks.

Example of usage	Press photographs are to be put in a unified format and published on the news platform on a daily basis.
Solution	Mistakes can be avoided and work can be completed faster and more efficiently by means of a predefined action defining the page relationship and the final image size.

Actions for batch processing

Actions combined with batch processing are the ideal solution for actions in the server area requiring a manipulation of pictures according to pre-set patterns. If applied to entire listings, such actions make it possible to process hundreds or thousands of pictures in no time at all. By creating cronjobs, the editing does not even need to be started manually.

A selection of example actions are available under /nexbyte/neximage/examples/actions. Simplified action invocations are available on http://neximage.host/examples/actions.

GUI actions

The action syntax is identical for GUI and for batch processing. GUI actions offer additional attributes, which disable the display of dialogue and settings.

Target users

These instructions are aimed at system administrators, technical staff and advanced system managers. Basic knowledge of XML (Extensible Markup Language) and experience in handling images is recommended.

Spelling

Standard values of attribute properties are written in italics.Please note: The XML-standard is case-sensitive, the same thing therefore applies to the interpretation of actions. The definition files need to be XML-compliant. When using the "&" sign within XML, please make sure that it is encoded as "&".

Action definitions are structured as follows:

```
<?xml version="1.0" encoding="iso-8859-1"?>
<!DOCTYPE neximage SYSTEM "http://neximage.com/dtd/neximage_action.dtd">
<neximage:action xmlns:neximage="http://neximage.com/ns/action/2.0/">

    ...

</neximage:action>
```

One or more commands are located in the "neximage:action" container near the stub "...". As regards the example codes listed in this document, only commands without the container are described.

In order for the actions to appear in the GUI selection, the file must be listed in the index. The index is located in the actionPath, which is /nexbyte/neximage/local/actions by default. Authorizations for executing actions can be limited at group level.

Invoking actions

Actions can be called up via shell or within the web browser. If no attributes are specified, usage indications are displayed with both invocation options. An alert will be displayed if the current user and his group lack sufficient authorization to read or write files and directories.

Shell

Invoking actions via the shell has the advantage of actions being processed under the current user authorizations. An invocation via the shell allows for an easier limitation of system resources.

/nexbyte/neximage/run/action/start in=world.jpg action=myaction.xml out=result.jpg

Web browser

Actions are executed with the authorizations of the user or group selected during installation. Please note that the query string may only contain URL-compliant signs.

http://neximage.host-bA/run/action/?in=world.jpg&action=myaction.xml&out=result.jpg

Arguments

Attribute	Description
license	The license as ID or alias.
share	Share resource for file reading/writing. The indication concerns "in", "out" and "action", if not defined more precisely.
in_share	Image reading resource (overrides "share")
out_share	Image writing resource (overrides "share")
action_share	Action opening resource (overrides "share")
workingdir	Working directory which is prefixed to relative paths. The indication concerns "in", "out" and "action", if not defined more precisely.
in_workingdir	Working directory which is prefixed to relative "in" paths (overrides "workingdir")
out_workingdir	Working directory which is prefixed to relative "out" paths (overrides "workingdir")
action_workingdir	Working directory which is prefixed to relative "out" paths (overrides "workingdir")
in	Base image to which the action is applied. A zip file may be used as an argument. Without specification, all archive images are edited. Individual archive images can be defined as follows: images.zip:img1.jpg
action	Action file name. First, the current directory, then the standard action directory is searched.

out	If only one "in" file is specified, "out" is the output file. If several "in" files are listed (e.g. *), then "out" must be a directory. If the base file is a ZIP archive, it is saved without the indication of "out" under the file with the same name. In this respect, please also note the "out_mode" argument. If "out_prefix" or "out_suffix" is specified, the argument may be left out. In order to overwrite the base image, you must specify the same name for "in" and "out". If "-" is indicated when processing an individual base image, the output to STDOUT is initiated.
out_prefix	Prefixes the name of an output file to a text
out_suffix	Adds the name of an output file to a text
out_mode	If "extract" is specified when processing a ZIP archive, the processed images are saved as individual files
format	Output image format (e.g. jpeg, png, ...). If not predefined, the image will be first selected analogically to the file extension (only possible if "out" is a file), otherwise the format of the base image is taken on.
quality	Output image quality. Can be set between 0 and 100. 100 stands for optimum quality, 0 for the worst quality and 75 refers to standard quality. The quality attribute is not supported by all image formats.
vars	Variables which are used for property substitution.
show	No specification by default. The following options are available: ▸ image_name Image name ▸ action_name action nametime Processing time per image ▸ image_in Base image (only accessible via web browser) ▸ image_out Resulting image (only accessible via web browser) ▸ totaltime Total processing time for all images and actions
onsuccess	Redirect URL which is invoked when no error has occurred
onerror	Redirect URL for an error-event. The error message is available as $NXI_ERROR variable.

Variables

Variables are used to influence the behavior of actions of invocation place and time. Variables are admissible for all action parameters. Variables are prefixed with the "$" dollar sign. If the variable name is not followed by a special character, the name has to be put between curly brackets. Example: ${vorname}test

Variable values are specified as follows on invocation: vars=var1:a,var2:b,...

Shell

When invoking via the shell, ensure that values containing spaces are put between inverted commas. Otherwise, the shell might misinterpret the arguments. Shell environment variables (predefined or inherent) are also available. When invoked, they are defined as follows: var1=a var2=b /nexbyte/neximage/run/action/start ...
If identical variables are available both as environment variables and within the argument "vars", those pertaining to the environment variables are overwritten.

Web browser

When invoking via the web browser, values should be URL-encoded. Equals signs in particular might cause problems.

Positioning

A great variety of properties are available for positioning layers. This ensures maximum flexibility when defining actions.

Attributes are set to x and y for positioning layers and for describing horizontal and vertical points. "x" refers to the distance between the top/left base point and the left border and
"y" refers to the distance to the top border. These specifications may contain negative values, and the layers may, under certain circumstances be situated outside the visible area in the event of an absolute positioning.

Base points for images

When a new layer is created, the coordinates (x/y) refer, by default, to the top left image corner. Therefore, the point x=0/y=0 is situated there. A positive offset moves the level to the right/bottom, a negative offset moves it to the left/top. Additional base points are required to place layers irrespective of their size and image dimensions.

Attribute	Description
hbase	Represents the horizontal base point. ▸ *left* *left* ▸ center centered ▸ right right
vbase	Represents the vertical base point. ▸ *top* *top* ▸ middle centered ▸ bottom bottom

9 different base points can be described by means of these two attributes.
Example:

hbase="right" vbase="bottom" means bottom/right or
hbase="center" vbase="middle" is the centre.

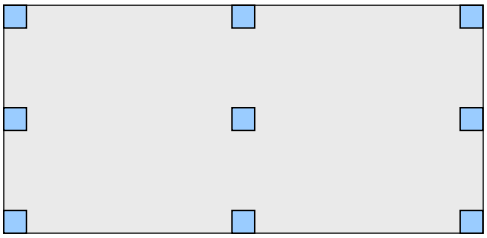
The orientation towards the base points may be defined freely for each layer per command.

Reference points for layers

The definition of base points only makes sense if the layer reference points can also be influenced. Only thus can for example a layer placed in the bottom right corner be scaled without it having to be moved from its predefined point, ten pixels away from the corner. If not otherwise defined, the base point definition is imposed on the reference point with the same value (inheritance of properties).

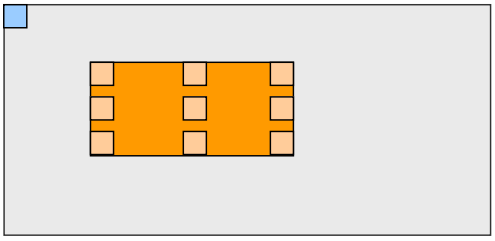
Attribute	Description
hreference	Represents the horizontal base point. ▸ *left* *left* ▸ center centered ▸ right right
vreference	Represents the vertical reference point. ▸ *top* *top* ▸ middle centered ▸ baseline Baseline (for text only) ▸ bottom bottom

Visual Display (Definitions)



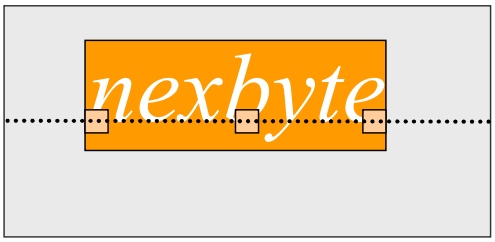
The grey bit-mapped background symbolizes the image surface. In this example, possible image base points are displayed in blue.

The base points have the attributes hbase and vbase.



The orange area symbolizes a layer which is positioned on the image surface. In this example, possible layer reference points are displayed in light orange.

The reference points have the attributes hbase and vbase.



The vertical reference point orientation for text levels can additionally be set to baseline. The vertical reference point thus aligns itself on the text baseline. Therefore, three additional reference points are available for text layers.

Visual Display (Examples)

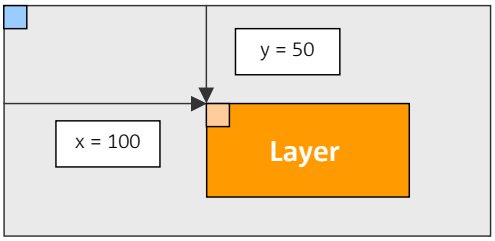


Image width:	240 px	hbase	left
Image height:	120 px	vbase:	top
Layer width:	100 px	hreference:	left
Layer height:	50 px	vbase:	top

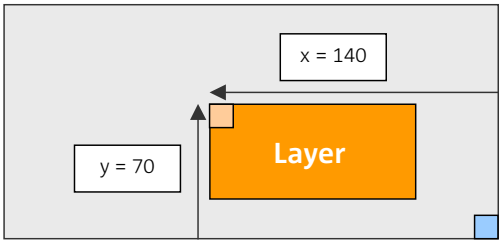


Image width:	240 px	hbase	right
Image height:	120 px	vbase:	bottom
Layer width:	100 px	hreference:	left
Layer height:	50 px	vreference:	top

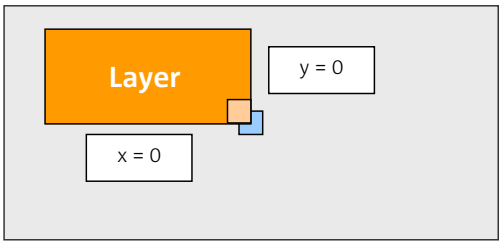


Image width:	240 px	hbase	center
Image height:	120 px	vbase:	middle
Layer width:	100 px	hreference:	right
Layer height:	50 px	vreference:	bottom

Commands

Crop (crop)

Both layers (target="layer") or entire images (target="image") can be cropped. The selection is made by means of width and height of the cropping frame and its starting position x and z. By means of an optional specification of the desired width (targetWidth) and height (targetHeight), the image or layer may be scaled to the indicated dimensions after cropping.

Attribute	Required	Description
target	no	▸ *image* ▸ layer
width	yes	width
height	yes	height
x	yes	position from left
y	yes	position from top
targetWidth	no	Target width After the selection has been cropped, the width is changed
targetHeight	no	Target height. After the selection has been cropped, the height is changed

Examples

Image cropping as of position 100, 20 with the dimensions 400x50.

```
<crop x="100" y="20" width="400" height="50" />
```

Scales the active layer to a maximum of 150x150 and moves to position 20.20.

```
<crop maxWidth="150" maxHeight="150" x="20" y="20" target="layer" />
```

Change canvas size (canvasSize)

CanvasSize is a simple tool for changing the work space dimensions. Use the attributes "width" und "height" to set new dimensions. The new work space is by default calculated from the centre outwards. Both a decrease and an increase are possible. Contrary to scaling (resize), layers keep their dimensions.

Attribute	Required	Description
width	yes	new width in pixels or percent (% adjusted)
height	yes	new height in pixels or percent (%adjusted)
align	no	▸ left left ▸ *center* *centered* ▸ right right
valign	no	▸ top top ▸ *middle* *centered* ▸ bottom bottom

Examples

Double canvas size (keep proportions) Centre existing layers.

```
<canvasSize width="200%" />
```

Reduce canvas size to 150x100 pixel. Base top left.

```
<canvasSize width="150" height="100" align="left" valign="top" />
```

Save image (saveOptimized)

The image resulting from an action is normally stored at the destination by means of the attribute"out". Should the current constellation by saved in the middle of an action, the command saveOptimized can be applied.

Attribute	Required	Description
path	yes	Destination path inclusive of file name. If the path is defined relatively (without the prefix "/"), the file is saved in the directory as defined in the "out" attribute. If this attribute has not yet been defined, the target file is saved in the action directory.
format	no	Output image format (e.g. jpeg, png, ...). If not predefined, the image format will be first chosen analogically to the file extension, otherwise the base image format is taken on.
quality	no	Output image quality. Can be set between 0 and 100. 100 stands for optimum quality, 0 for the worst quality and 75 represents standard quality. The quality is not supported by all image formats.
colors	no	Number of colors for saving an indexed image. This option may only be used when format="gif" is set. Possible values are located between 1 and 256.
dither	no	Regulates dithering of indexed images. ▸ *false* without dithering ▸ true activate dithering

Scale (resize)

Scale image or layer. If width and height are set, the target is scaled absolutely. Width and height might also be left out. This makes it possible to scale proportionally. Maximum width (maxWidth) and maximum height (maxHeight) make it possible to reduce the size of the target proportionally. Only one value is required for maximum dimension indications. When target="layer" has been selected, an additional layer move can be specified by means of x/y.

Attribute	Required	Description
target	no	▸ *image* ▸ layer
width	no	absolute width in pixel or percent (% adjusted)
height	no	absolute height in pixel or percent (%adjusted)
maxWidth	no	maximum width
maxHeight	no	maximum height

x	no	horizontal layer move after scaling (only possible with target="layer")
y	no	vertical layer move after scaling (only possible with target="layer")

Examples

Image scaling to target size, 100x200 dimensions. The resulting image may be distorted.

```
<resize width="100" height="200" />
```

Proportional image scaling to a width of 100 pixel. Automatic height adjustment.

```
<resize width="100" />
```

Proportional scaling of the active layer to a maximum of 150x150 and move to position 20.20.

```
<crop maxWidth="150" maxHeight="150" x="20" y="20" target="layer" />
```

Rotate (rotate)

Image or layer rotation in 90° steps. For counter-clockwise rotations a minus sign is prefixed.

Attribute	Required	Description
target	no	▸ *image* ▸ layer
angle	yes	Rotation in 90° steps

Examples

Rotate image clockwise by 90°.

```
<rotate angle="90" />
```

Rotate image counter-clockwise by 270° (identical to previous example).

```
<rotate angle="-270" />
```

Rotate active layer by 180°.

```
<rotate angle="180" target="layer" />
```

Flip (flip)

Flip image or layer horizontally or vertically.

Attribute	Required	Description
target	no	▸ *image* ▸ layer
direction	yes	▸ *horizontal* ▸ *vertical*

Examples

Flip image horizontally

```
<flip />
```

Flip active layer horizontally.

```
<flip direction="vertical" target="layer" />
```

Add layer (addLayer)

An image or text layer is added to the image. If no hierarchy position (above/below) is selected, the newest layer is added at the top. Without horizontal and vertical position specification, the newest layer appears in the top left corner.

Attribute	Required	Description
type	no	▸ *image* ▸ text
name	no	Name of the layer used for referencing. If not specified, the name "layer x" is selected, x being the position within the hierarchy.
above	no	Position above the specified layer (name).
below	no	Position below the specified layer (name).
x	yes	X-coordinate for the horizontal position. With regard to text layers, the coordinate refers to the horizontal alignment (align)
y	yes	Y-coordinate for the vertical position. With regard to text layers, the coordinate refers to the baseline of the first text line
hbase	no	vertical image base point
vbase	no	horizontal image base point
hreference	no	horizontal layer reference point
vreference	no	vertical layer reference point

opacity	no	Opacity. The value can either be specified as 0.01-1.0 or as a percentage 0-100%. Percentage values must be suffixed with %. The default value is 100%.
visible	no	Layer visibility ▸ *true* *Layer is visible* ▸ false Hide layer

Image layer specific (type="image")

path	yes	Image file path

Text layer specific (type="text")

fontFamily	yes	Font type (Arial, Times, ...)”
fontStyle	yes	Font style (italic, bold, ...)
fontSize	yes	Text size
color	no	Text color as a hexadecimal value. Standard color is black (#000000).
string	yes	text
align	no	▸ *left* *left-aligned* ▸ center centered ▸ right right-aligned

Examples

Create new image layer below the layer with the name "Layer3".

```
<addlayer type="image" name="Ebene2" path="image.png" below="Ebene3" x="10" y="20" />
```

Create new text layer at the top. The name is set in accordance with the position (layer n).

```
<addlayer type="text" fontFamily="Arial" fontStyle="bold" fontSize="22" color="#ff9900"
string="mytext..." x="5" y="100" />
```

Reposition layer coordinates (setLayerPosition)

Layers may be repositioned horizontally and/or vertically in absolute or relative terms. The active layer is moved. One of the attributes "x" or "y" must be specified.

Attribute	Required	Description
position	no	▸ *absolute* *absolute positioning* ▸ relative relative move
x	no	absolute coordinate for the horizontal layer or distance positioning with regard to the relative pixel offset.
y	no	absolute coordinate for the vertical layer or distance positioning with regard to the relative pixel offset.
hbase	no	vertical image base point
vbase	no	horizontal image base point
hreference	no	horizontal layer reference point
vreference	no	vertical layer reference point

Example

Move layer by 20 pixel horizontally and 50 pixel vertically

```
<setLayerPosition x="20" y="50" position="relative" />
```

Arrange layer sequence (moveLayer)

The layer sequence affects overlapping and semi-transparent images. As opposed to moving layer coordinates, the vertical and horizontal positioning remains the same. The change affects the currently selected layer.

Attribute	Required	Description
switch	no	▸ up one layer backwards ▸ down one layer forwards ▸ top place in foreground (last layer) ▸ bottom place in background (first layer)
above	no	Move behind the layer with the specified name
below	no	Move in front of the layer with the specified name

Example

Place the active layer above the "grid" layer.

```
<moveLayer above="grid" />
```

Select layer (selectLayer)

Change active layer. The selection takes place by means of name or position.

Attribute	Required	Description
name	no	Name of the layer to be selected. Case-insensitive.
number	no	Position of the layer to be selected. The lowest layer has position 1.

Example

Select layer with the name "Sonne". This may be followed by any amount of actions referring to the active layer.

```
<selectLayer name="Sonne" />
```

Delete layer (deleteLayer)

Delete a layer. If neither name nor position is specified, the active layer is deleted. Once the active layer is deleted, the level below is activated.

Attribute	Required	Description
name	no	Name of the layer to be deleted. Case-insensitive.
number	no	Position of the layer to be deleted. The lowest layer has position 1.

Example

Delete active layer.

```
<deleteLayer />
```

Copy layer (copyLayer)

Copy active layer. The copy may be given the attribute "name". If no name is defined, the layer is given a name in the form of "layer n", with n standing for the position.

Attribute	Required	Description
name	no	Name of the layer copy

Example

Copy active layer and name as "copied layer".

```
<copyLayer name="kopierte Ebene" />
```

Combine layers (flatten)

All the available layers are combined into one single layer. The new layer is given the name of the lowest layer. There are no attributes available for this command.

Example

```
<flatten />
```

Filter (filter)

Application of filters to the active layer. The filter type is indicated with "prop".

Color correction

At least one of the attributes brightness, saturation and color must be indicated.

Attribute	Value	Required	Description
type	modulate	yes	
brightness	n	no	Brightness 0 Minimum level (black) 1-99 darker 100 Standard (no change) > 100 lighter
saturation	n	no	Saturation 0 no color < 100 Reduce saturation 100 Standard (no change) > 100 Increase saturation
color	n	no	Color (shade) < 100 Reduce color 100 Standard (no change) > 100 Increase color

Contrast

Attribute	Value	Required	Description
type	contrast	yes	
value	n	yes	< 0 less contrast 0 Standard (no change) > 0 more contrast

Autocontrast adjustment

Attribute	Value	Required	Description
type	autocontrast	yes	

Invert

Attribute	Value	Required	Description
type	negate	yes	

Grayscales

Attribute	Value	Required	Description
type	grayscale	yes	

Sharpen

Attribute	Value	Required	Description
type	sharpen	yes	
value	#f	yes	Standard deviation in pixel

Blur

Attribute	Value	Required	Description
type	blur	yes	
value	#f	yes	Standard deviation in pixel (decimals possible)

Spread

Attribute	Value	Required	Description
type	spread	yes	
value	#d	yes	Radius as an integer

Swirl

Attribute	Value	Required	Description
type	swirl	yes	
value	#d	yes	Angle of twist (-999 to 999) > 0 swirl right < 0 swirl left

Change properties (setProp)

The properties of an image or the active layer are changed. The following properties may be changed:

Attribute	Layer type	Description
target		▸ imagelayer
opacity	image	The value can either be specified as 0.0-1.0 or as a percentage 0-100%. Percentage values must be suffixed with %. The standard value is 100%
visible	no	Layer visibility ▸ true *Layer is visible* ▸ false Hide layer
fontFamily	text	Font type (Arial, Times, ...)
fontStyle	text	Font style (italic, bold, ...)
fontSize	text	Text size
color	text	Text color as a hexadecimal value. Standard color is black (#000000).
string	text	text
align	text	▸ left left-aligned ▸ center centered ▸ right right-aligned

Specific GUI matters

The commands listed above may be used both for batch processing and within the GUI. The following commands may only be invoked within the GUI:

Command	Description
setResizeTool	Activate scaling tools. Standard attributes may be set analogically to scale (resize).
setCropTool	Activate crop tool. Standard attributes may be set analogically to crop (crop).
setMoveTool	Activate move tool
setTextTool	Activate text tool
setPanTool	Activate pan tools. This makes it possible to move the work space. The scroll bars offer similar functionalities

When used within the GUI, the way in which the user is able to influence the workflow and the values can be defined for each commands. The following attributes are available:

Attribute	Required	Description
disabled	no	List of attributes which may not be altered. A space is used as the list separator
interactive	no	▸ false *Skip dialogue* ▸ true Show dialogue (if available)

If interactive is set to "true", each command may by default be cancelled by pressing the ESC-key or clicking on the cancel button. For individual handling, the attribute onCancel may be set within the Root Element "neximage:action". This makes it possible to change the behavior in this particular case.

Attribute	Required	Description
onCancel	no	▸ skip *Skip command* ▸ *stop* Stop action ▸ *undo* Stop action and reverse commands already executed

Extended Examples

The directory /nexbyte/neximage/examples/actions/advanced lists extended action examples. The individual examples are described in detail below.

add_multiple_layers_*.xml

This makes it possible to add four identical image layers around the image centre. As there is a center-orientation, the action does not depend on the base image size.

There are several ways of adding several image layers with the same path. Each of these is identical with regard to memory requirements and speed.

Version	Description
1	Add layers by means of Add layer (addLayer) and the same path
2	Add the first layer by means of Add layer (addLayer), copy all further ones and move to the correct position by means of move layer coordinates

corner_text.xml

In order to position a changeable text layer on a fixed image position, the text has to be changed before the position can be set. This is due to the fact that the base and reference point specification only refers to one command and is not kept permanently.

center_text.xml

In order to add a text to the centre of an image, the base point is set horizontally to center and vertically to middle. Thus, the standard values for the layer reference, which align themselves horizontally on the text orientation and vertically on the base line, are overwritten.

dynamic_text.xml

The use of variables makes it possible to calculate dynamic texts on images. Variables, for example, may be set directly by return values of command line programs. Execute the action by means of the following variable definition: vars="date:`date +%H:%M:%S`, sport:swimming"
By copying and moving the text layer by 1 pixel, a shadow is additionally attached to text.

filter_area.xml

Filters are always applied to the entire layer. If only a small part of a layer is to be edited, there is a trick you may use: Copy the layer that needs to be edited by means of Copy layer (copyLayer), crop the area by means of Crop (crop) and apply the filter to the layer.