unternehmen online

WebFlowChart SDK

Sascha Jung
Unternehmen Online GmbH & Co. KG

Project: WebFlowChart 0.8.8

Version: 1.0 (Beta)
Created: 12.11.2003
Last Change: 09.06.2004

Content

1. INTRODUCTION

Like any other application **WebFlowChart** started with an idea. This idea was to integrate the ability to create, edit and save Flow Chart graphs in a Web environment like an Intranet/Internet Content Management Application.

To stay platform independent and ensure Browser interoperability we decided to develop **WebFlowChart** in Java 1.1 as applet.

2. REQUIREMENTS

The following requirements are the minimal recommended environment.

2.1. Client

2.1.1. Java™ VM

WebFlowChart needs at least a Sun Java **1.1 compatible Java™ VM**

2.1.2. System

To run proper **WebFlowChart** needs at least an about 300Mhz PC with 64MB RAM or any other Java™ 1.1 capable comparable system.

Note: This is just a recommendation. Different Java™ VM implementations run faster than the original Sun VM.

2.2. Server

WebFlowChart runs with any http multipart/form request capable webserver.

3. INSTALLATION

The component inherently itself needs no installation, but it has to be implemented in your environmental system.

3.1. Client

The **WebFlowChart** applet is divided into two parts which all are located in the `com.uo_components.webflowchart` package.

An applet naturally does not have to be installed on the client system, but you have to ensure that this applet is delivered. A single **WebFlowChart** applet will open an editor window after loading all resources or (depends on the configuration) wait until it is triggered by a **TriggerButton** applet.

With this SDK sample implementations are delivered.

3.2. Server

To load and save files **WebFlowChart** needs two server side scripts which have to be implemented concerning to the environmental system.

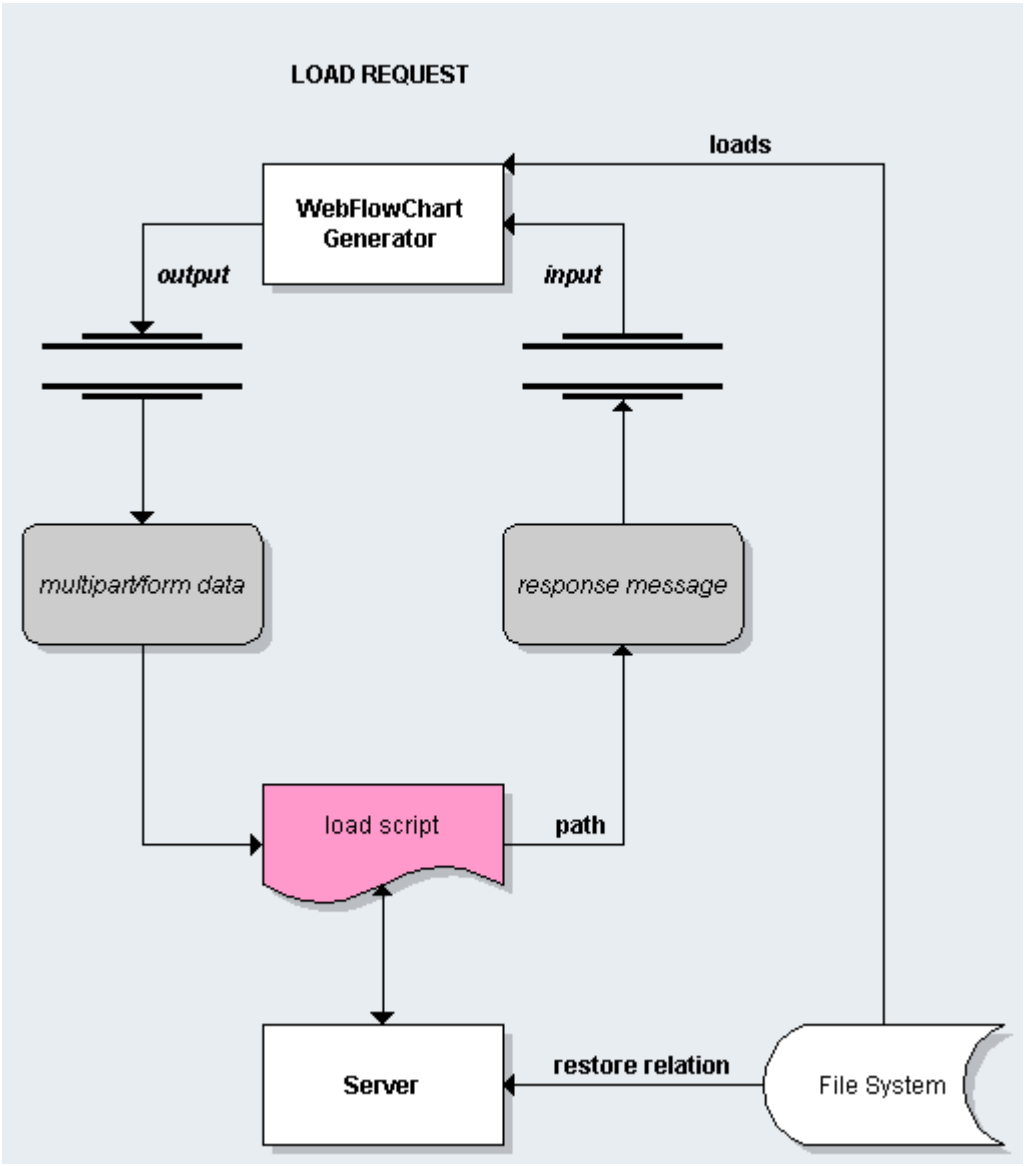
Sample implementations are part of this SDK

3.2.1. Save script

To save EPF-files and Flowchart Graphics **WebFlowChart** needs a save script.

On user "save request" the editor will send a `multipart/form http post request` to the save script.

The following scheme shows a typical save process.



The form request can contains the following fields:

Field Name	Description	MIME type
ID	If this value is "**-1**" the submitted flow chart is "new" and should not override any other previous file. Otherwise this is the current used flow chart id.	text/plain
EPF	Contains the EPF-File binary. NOTE: This field won't be submitted when not specified in the export formats param.	application/epf
PNG	Contains the PNG-File binary. NOTE: This field won't be submitted when not specified in the export formats param.	image/x-png
Title	Contains the flow chart title. NOTE: This field won't be	text/plain

	submitted if the document title has not been set.	

After handling the save request the **WebFlowChart Editor** awaits a response from the `save script`, which can either be an error message or a new document id.

If for some reason an error occurred the `save script` may notify the applet by returning an error description prefixed with "**Error:**" (e.g. `"Error: Server could not save file"`).

PHP error sample

```
$errorid = "Ooops… something is wrong here";
echo "Error:".$errorid;
```

The **WebFlowChart Editor** will show this error message to the user.

Anything else returned will be used as new document id. So the **WebFlowChart Editor** will use the received "new" id as **"ID"**.

This way you could easily realize a flow chart versioning mechanism.

NOTE: The `save script` **must** return at least the old id!

The following script shows you how to implement a simple save script.

Sample PHP save script code

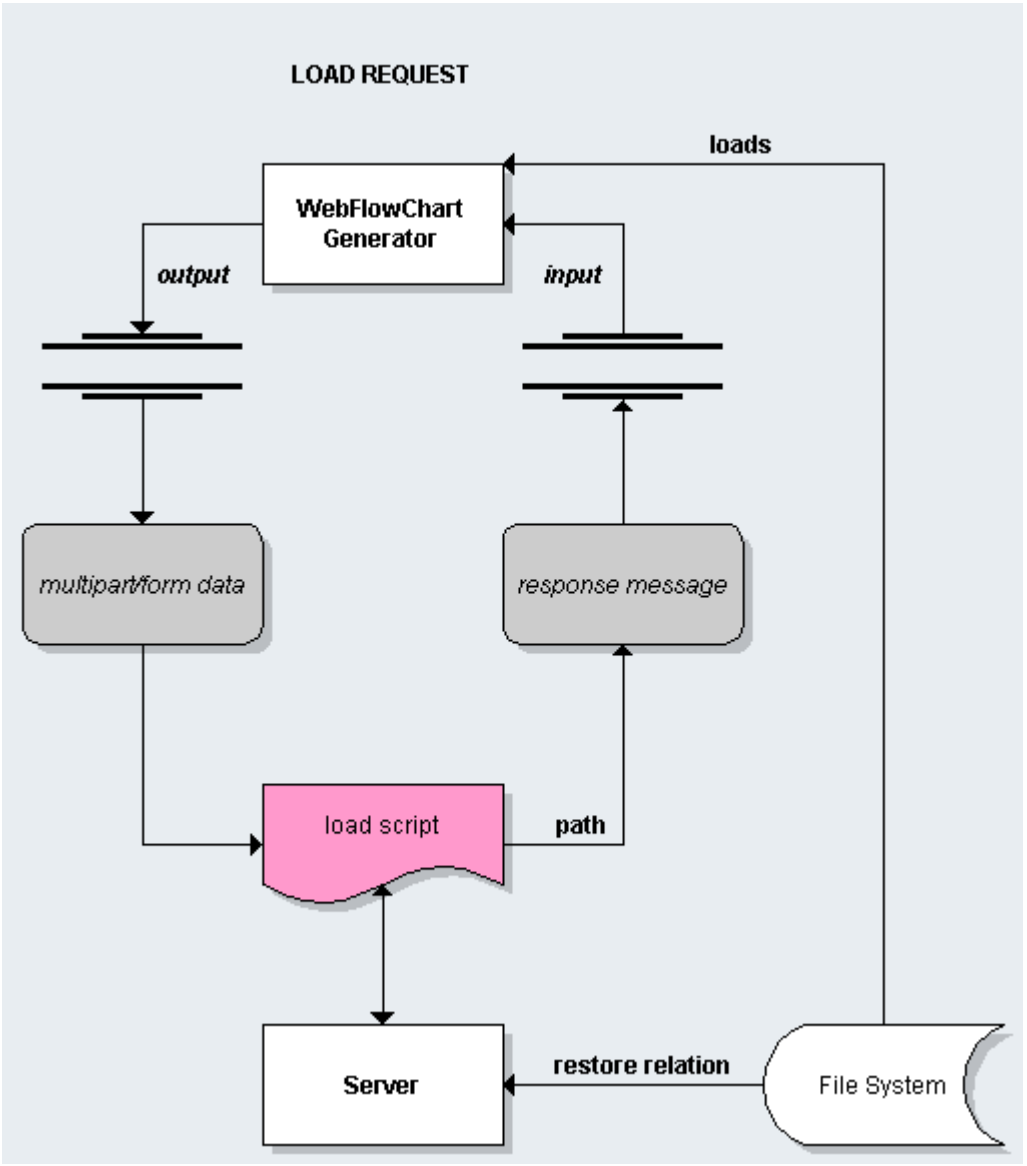
```
<?
#-------------------------------------------------------------#
# This example script is appropriate for PHP 4.1.0 or higher. #
#-------------------------------------------------------------#
// Checks whether id exists.
if(!isset($_POST["ID"])) {
   exit();
}
// Checks whether id was already setted,
// otherwise set it to unix timestamp.
$id = $_POST["ID"];
if($id == -1) {
   $id = "epf_".time();
}
// Save your id into a session, if you want to use it in the other sites.
session_start();

// Prepares directory for uploaded files.
$physPath = substr($_SERVER["SCRIPT_FILENAME"], 0, strrpos($_SERVER["SCRIPT_FILENAME"], "/") +
1);
$uploaddir = $physPath."datastore/";

// Saves epf file.
// Note, that post file-format params are uppercase,
// for exact param declaration, use SDK.
if($_FILES["EPF"]) {
   if(!SaveFile("EPF", addslashes($uploaddir).$id.".epf")) {
      echo "Error:Error occured during EPF save.";
      exit();
   }
}


// Saves png file.
if($_FILES["PNG"]) {
   if(!SaveFile("PNG", addslashes($uploaddir).$id.".png")) {
      echo "Error:Error occured during PNG save.";
      exit();
   }
}

// finally succesful!
echo $id;

function SaveFile($fileFormat, $path){
    return move_uploaded_file($_FILES[$fileFormat]['tmp_name'], $path);
}
?>
```

3.2.2. Load script

When a **WebFlowChart Editor** was started with the id "-1", it will send a `multipart/form http post request` to the load script requesting the id related document path.

(Naturally a new file does not have to be loaded from the Server before even created)

The following scheme shows a typical load process.



The form request contains the following fields:

Field Name	Description	MIME type
ID	Contains the flow chart ID.	text/plain

The `load script` now has to return a string containing the relative path (from the editor applets codebase) to the EPF-file that is related with this id (e.g. `"/data/myflowchart.epf"`).

If for some reason an error occurred the `load script` may notify the applet by returning an error description prefixed with "**Error:**" (e.g. `"Error: Server could not find file"`).

The **WebFlowChart Editor** then will show this error message to the User.

The following script shows you how to implement a simple load script.

Sample PHP load script code

```
<?php
#-------------------------------------------------------------#
# This example script is appropriate for PHP 4.1.0 or higher. #
#-------------------------------------------------------------#
// Checks whether id exists.
if(!isset($_POST["ID"])) {
   exit();
```

```
}
// Loads epf file within the specified id.
echo "../samples/php/datastore/".$_POST["ID"].".epf";
?>
```

4. CONFIGURATION

The component can be easily configured by referring to the following descriptions.

4.1. Applet

You can configure the behavior of **WebFlowChart** with setting the params of the applet

4.1.1. com.uo_components.webflowchart.Starter

This main application applet class loads all needed resources and opens a editor window or waits for trigger events from a **TriggerButton**.

Param Name	Description	Default value
debuginfo	If the "**debuginfo**" param is set to true the **WebFlowChart** will start up with special developer features. Possible values are "true" and "false"	false
savepage	Sets the relative path from the applets codebase to the save requests processing script.	"/save.asp"
loadpage	Sets the relative path from the applets codebase to the load requests processing script.	"/load.asp"
imagepath	Sets the relative path from the applets codebase to the image resource folder	"images"
id	When the **Starter** trigger should open a previous saved document this param indicates the document id. Anything else than "-1" will be accepted as document id. "-1" is the default id for a new document.	-1
autotrigger	If the applet should trigger itself after loading all resources with the "**id**" param instead of waiting for a trigger event from a **TriggerButton** Possible values are "true" and "false"	false
submitformats	A comma separated export formats list which will be submitted with save (e.g. If you only want the editor to submit the png fill this param with "png". If you want it to submit png and epf fill this param with "epf,png") Current supported export formats are: • epf • png	none
urlafterclose	If the **WebFlowChart** should load a URL after closing an Editor. If you want that the last ID that was returned from the savepage should be submitted to this URL, use the "`{%id_placeholder%}`" which will be replaced with the last returned ID. http://www.foobar.com?id=`{%id_placeholder%}`	""
urlafterclosetarget	Must be set if "**urlafterclose**" is given. This param defines the target frame into which the after save URL should be loaded. (e.g. "_self")	""

loadingtext	The Text which is shown in the loadingbar while loading the resources.	“WebFlowChart”
readytext	The text which is shown in the loadingbar after loading the resources.	“WebFlowChart”
borderhighlightcolor	Sets the HEX encoded border highlight RGB color.	“F2F2F2”
bordershadowcolor	Sets the HEX encoded border shadow RGB color.	“354F66”
headcolor	Sets the HEX encoded RGB color for the editor logo head.	“354F66”
fontcolor	Sets the HEX encoded RGB color for the editor font.	“344E65”
bgcolor	Sets the HEX encoded main background RGB color.	“ADBACE”
fgcolor	Sets the HEX encoded main foreground RGB color.	“E8EDF1”
languageid	Sets the applications language using ISO 639-1 codes for the representation of names of languages Possible current supported values are “de” for German and ”en” for English	“de”
documentwidth	Sets the default width for new documents	800
documentheight	Sets the default height for new documents	600
documentbgcolor	Sets the default document background color for new documents	**fgcolor**

Application applet sample HTML/PHP Code

```
<applet codebase="../../bin/" code="com.uo_components.webflowchart.Starter.class"
archive="webflowchart.jar" name="" width="290" height="20" hspace="0" vspace="0"
align="center">
    <PARAM NAME="debuginfo" VALUE="false">
    <PARAM NAME="savepage" VALUE="../samples/php/savepage.php">
    <PARAM NAME="loadpage" VALUE="../samples/php/loadpage.php">
    <PARAM NAME="imagepath" VALUE="../skins/default/images">
    <PARAM NAME="autotrigger" VALUE="false">
    <PARAM NAME="submitformats" VALUE="epf,png">
    <PARAM NAME="urlafterclose" VALUE="<?echo
"http://".$_SERVER["HTTP_HOST"].$_SERVER["SCRIPT_NAME"]."?closed=1&flowChartId={%id_placeholde
r%}";?>">
   <PARAM NAME="urlafterclosetarget" VALUE="_self">
</applet>
```

4.1.2. com.uo_components.webflowchart.TriggerButton

This is just a trigger applet class that, on mouseclick, will open a new **WebFlowChart** Editor with the file associated with this buttons ID.

The trigger button stays disabled until the **Starter** class has loaded all images from the server.

Param Name	Description	Default value
id	When the **TriggerButton** should open a previous saved document this param indicates the document id. Anything else than “-1” will be accepted as document id. “-1” is the default id for a new document.	-1
imagepath	Sets the relative path from the applets codebase to the image resource folder	“images”

Trigger applet sample HTML Code

```
<applet codebase="../../bin/" code="com.uo_components.webflowchart.TriggerButton.class"
archive="webflowchart.jar" name="" width="32" height="32" hspace="0" vspace="0" align="right">
    <PARAM NAME="id" VALUE="-1">
    <PARAM NAME="imagepath" VALUE="../skins/default/buttonimages">
```

```
</applet>
```

4.2. Skin

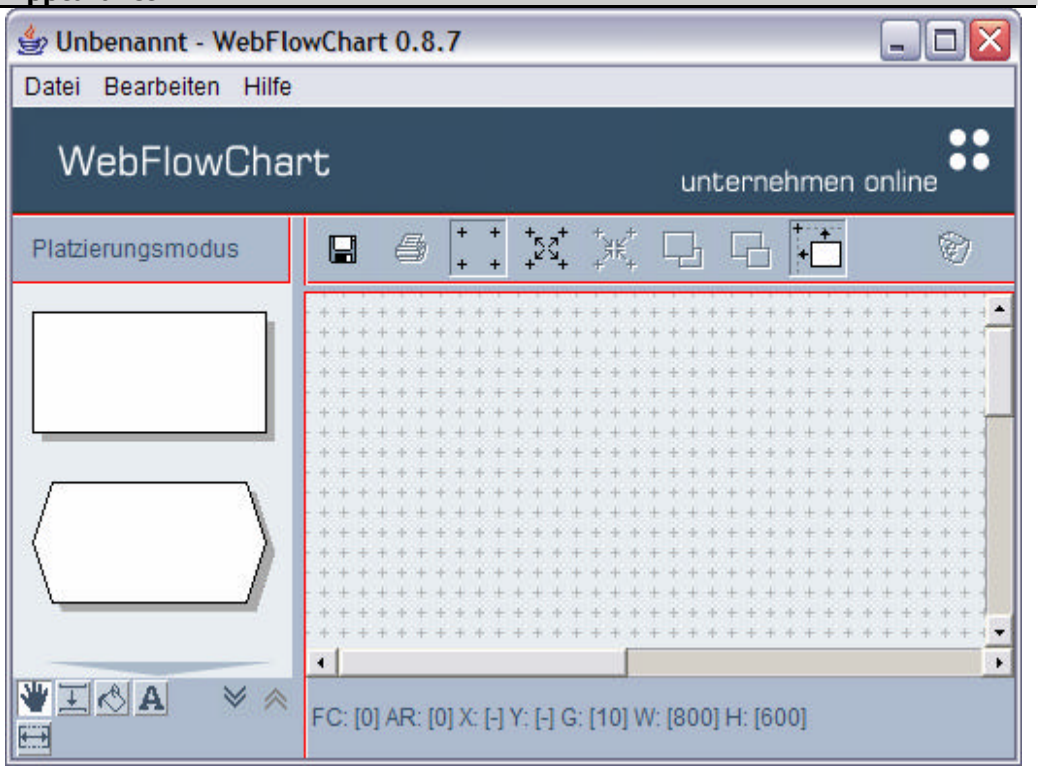
WebFlowChart is a fully skinable component.

You can use one of our default skins or create your own by replacing the images and setting the color params in the `com.uo_components.webflowchart.Starter` applet.

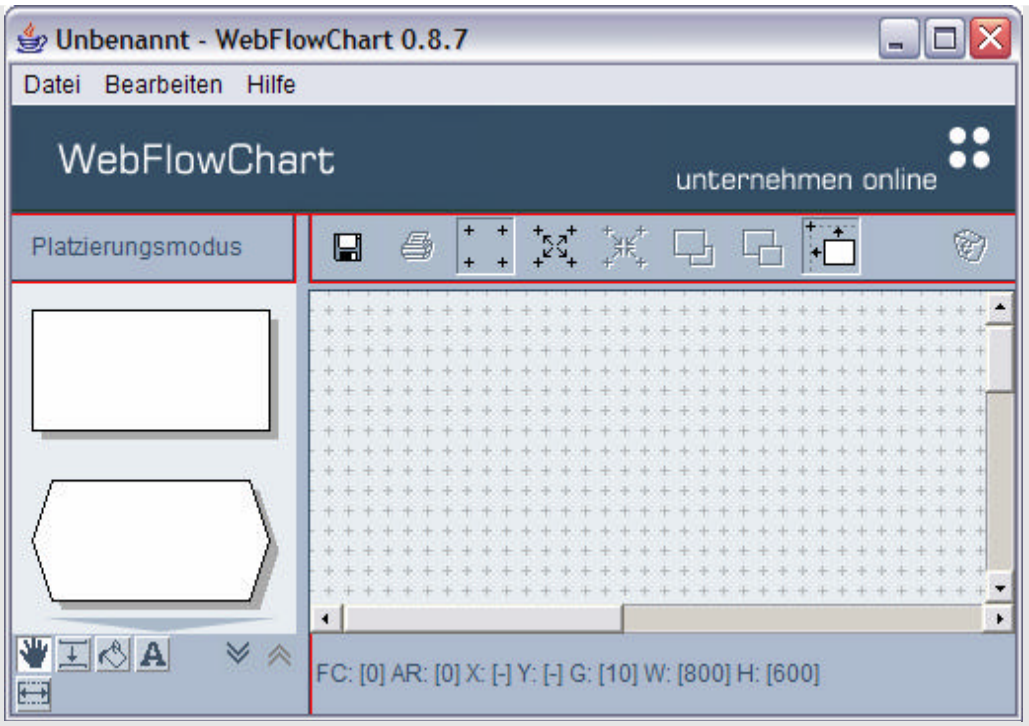
4.2.1. Colors

In the table below all editable colors are listed and shown in red in the appearance preview.

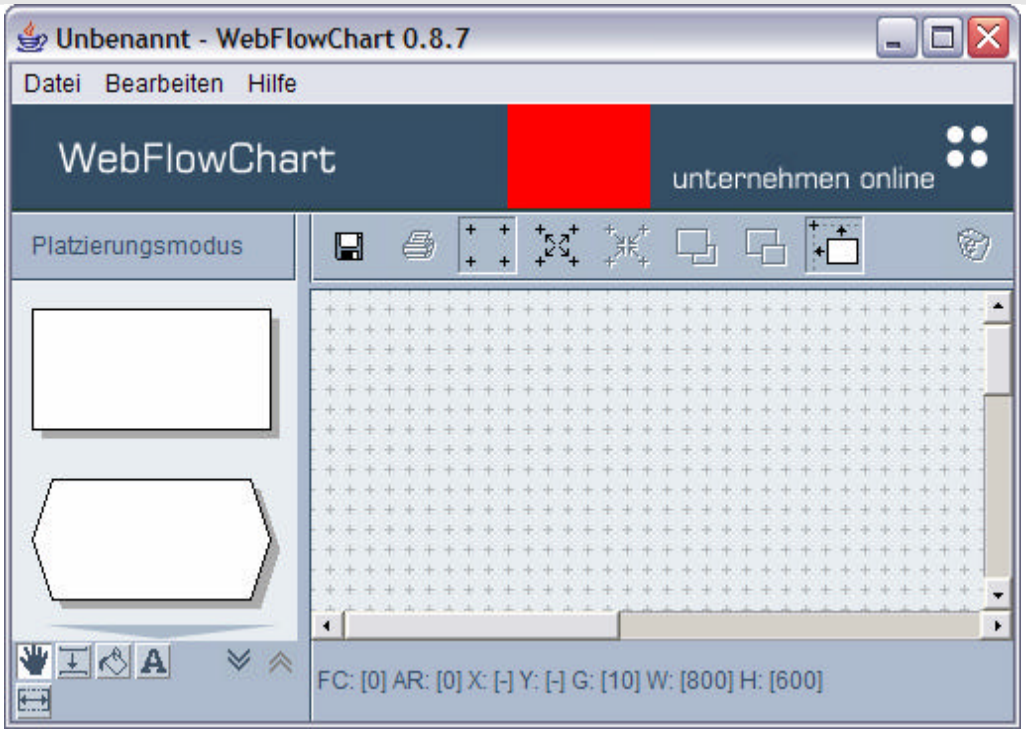
In **/skins/fiktiv** in this SDK you can find a sample implementation of a different skin.

Color param	Appearance
bordershadowcolor	

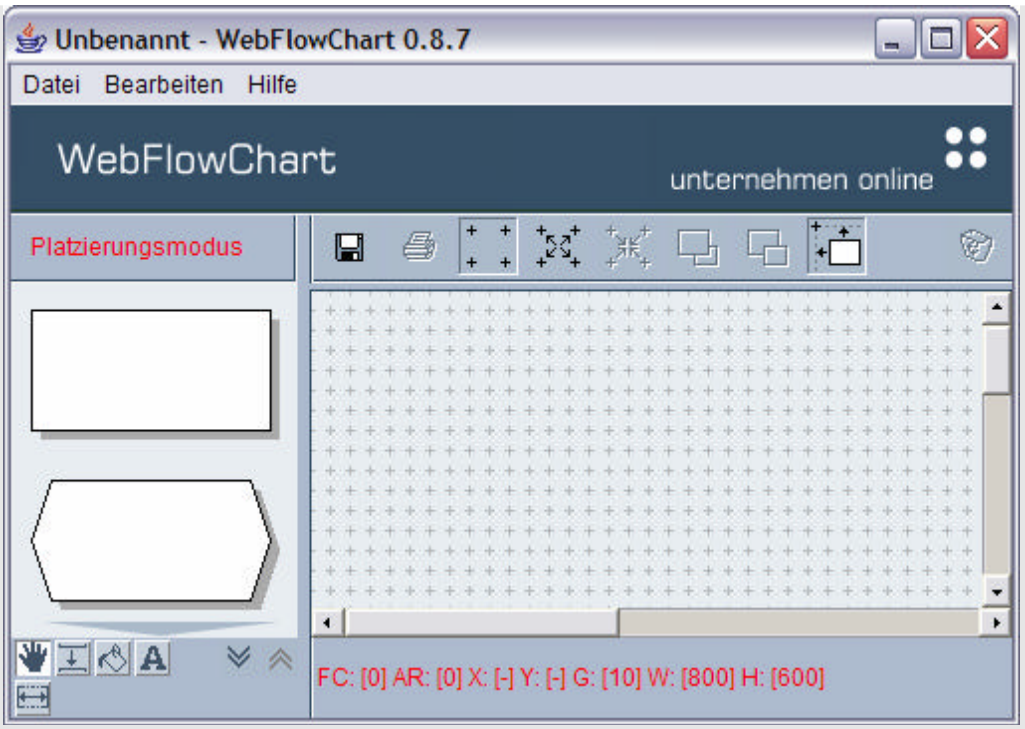
borderhighlightcolor



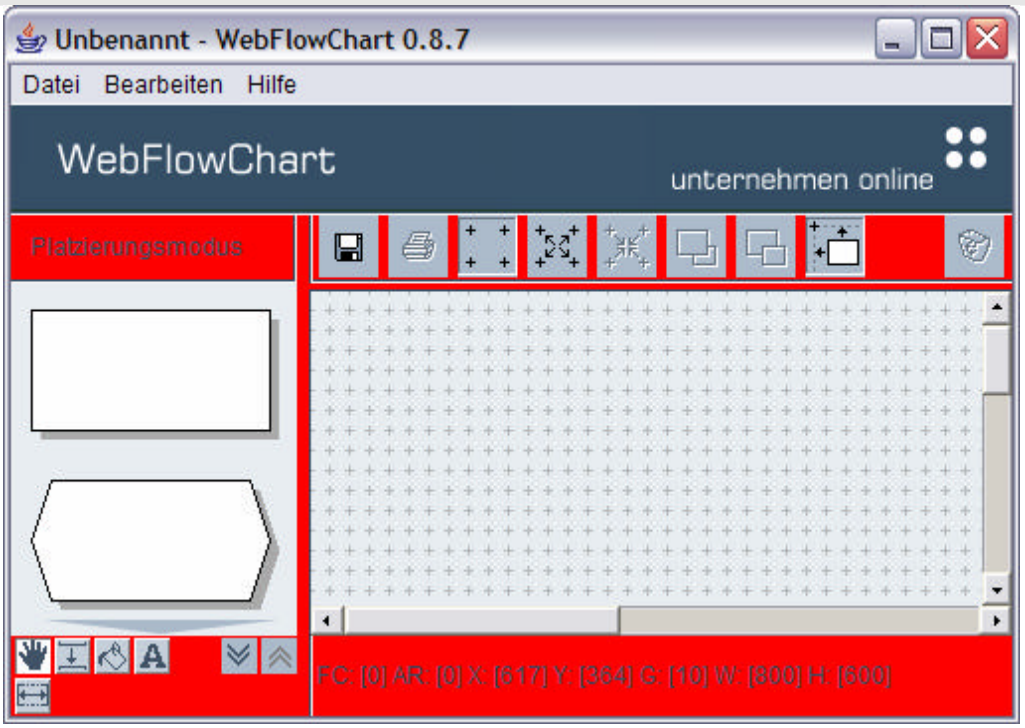
headcolor

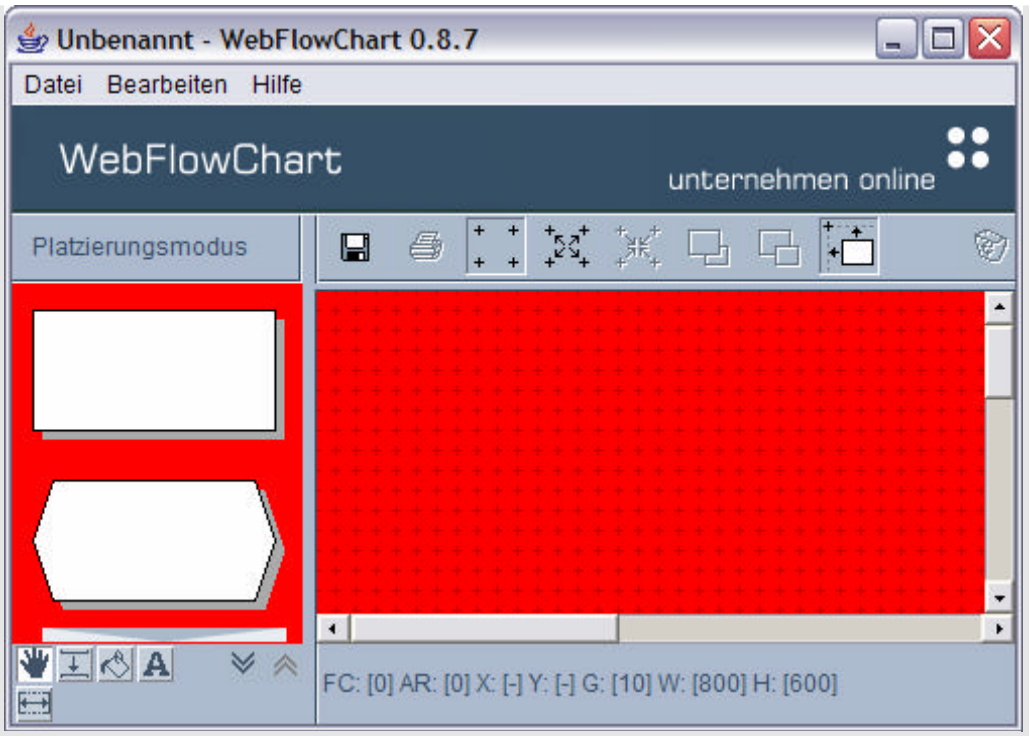


fontcolor



bgcolor



fgcolor	 Unbenannt - WebFlowChart 0.8.7 Datei Bearbeiten Hilfe WebFlowChart unternehmen online Platzierungsmodus FC: [0] AR: [0] X: [-] Y: [-] G: [10] W: [800] H: [600]

5. APPENDIX

5.1. Terms

Term	Description
EPF	**ePilot Flowchart File Extension** **WebFlowChart** uses this file format to store the Flow Chart documents data.
PNG	**Portable Network Graphics File Extension** Image file format used by the **WebFlowChart Editor** to export an EPF-file.

5.2. Known Problems

5.2.1. Netscape 4.x

Under several Versions of Netscape 4.x **WebFlowChart** cannot correctly produce an PNG caused in a major bug in the Virtual Machine.

In this case we recommend to install a newer Version of Netscape.