hyper
☐ Content & Digital Asset Management Server
☐ CMS

Version 7.0 Programers Guide

Inhalt

1	Einleitu	ung	1
2	hyperC	CMS XML-Content-Repository	1
		yperCMS spezifische Informationen	
		eta-Informationen	
		ext	
		edien	
		nks	
		omponenten	
		rtikel	
3		onsbibliotheken	
J		nbindung einer Bibliothek	
		aden der Konfiguration	
	3.2.1 3.2.2	Content Management Server	
		Publication Server	
		lobale Variablen	
		orlagenvariablen	
		bliothek Object Operation	
	3.5.1	createfolder	
	3.5.2	deletefolder	
	3.5.3	renamefolder	
	3.5.4	createobject	
	3.5.5	deleteobject	
	3.5.6	renameobject	
	3.5.7	cutobject	
	3.5.8	copyobject	18
	3.5.9	copyconnectedobject	19
	3.5.10	pasteobject	20
	3.5.11	lockobject	21
	3.5.12	unlockobject	22
	3.5.13	publishobject	23
	3.5.14	unpublishobject	24
	3.5.15	getlinkedobject	25
	3.5.16	getconnectedobject	26
	3.5.17	getobjectcontainer	27
	3.5.18	loadcontainer	27
	3.5.19		
	3.6 Bi	bliothek File Pointer	29
	3.6.1	getfilenamegetfilename	
	3.6.2	setfilename	
	3.7 Bi	bliothek File Operation	
	3.7.1	loadfile'	
	3.7.2	savefile	
	3.7.3	loadlockfile	
	3.7.4	savelockfile	
	3.7.5	lockfile	
	3.7.6	unlockfile	
	3.7.7	deletefile	
	3.7.8	appendfile	
		ibliothek Edit Content	
	3.8.1	setxmlparameter	
	3.8.2	getcontent	
	3.8.3	getxmlcontentgetxmlcontent	
	3.8.4	selectcontent	
	3.8.5	selectxmlcontent	
	3.8.6	deletecontent	
	5.0.0	uoiotoouttoitt	ΨU

	3.8.7	setcontent	41
	3.8.8	updatecontent	42
	3.8.9	insertcontent	43
	3.8.10	addcontent	44
	3.9 Bib	oliothek Meta Data Generator	45
	3.9.1	getkeywords	45
	3.9.2	getdescription	45
	3.9.3	injectmetadata	46
	3.10 Bib	oliothek Notifications	
	3.10.1	licensenotification	
4	Kompor	nenten und Applikationen	48
5		se Connectivity	
		stellen einer Database Connectivity	
6		ystem	
7		r hyperCMS API Funktionen	
		in API Functions	
	7.1.1	correctnumber	
	7.1.2	cleancontent	
	7.1.3	remove_utf8_bom	
	7.1.4	convertchars	
	7.1.5	specialchr	
	7.1.6	specialchr_encode	
	7.1.7	specialchr_decode	
	7.1.7	object_exists	
	7.1.8 7.1.9	is_utf8	
	7.1.9 7.1.10		
		is_latin1	
	7.1.11	makestring	
	7.1.12	splitstring	
	7.1.13	is_emptyfolder	
	7.1.14	is_supported	
	7.1.15	is_cloudstorage	
	7.1.16	is_cloudobject	
	7.1.17	is_date	
	7.1.18	is_tempfile	
	7.1.19	is_thumbnail	
	7.1.20	is_config	
	7.1.21	is_encryptedfile	
	7.1.22	is_document	
	7.1.23	is_image	
	7.1.24	is_rawimage	
	7.1.25	is_aiimage	
	7.1.26	is_video	
	7.1.27	is_audio	
	7.1.28	is_mobilebrowser	
	7.1.29	is_iOS	60
	7.1.30	copyrecursive	60
	7.1.31	array_iunique	61
	7.1.32	in_array_substr	61
	7.1.33	createfilename	61
	7.1.34	correctfile	62
	7.1.35	correctpath	62
	7.1.36	convertpath	62
	7.1.37	convertlink	
	7.1.38	deconvertpath	
	7.1.39	deconvertlink	
	7.1.40	createviewlink	
	7.1.41	createaccesslink	
	7.1.42	createobjectaccesslink	
	· · · · · -		

7.1.43	createwrapperlink	65
7.1.44	createdownloadlink	
7.1.45	createmultidownloadlink	
7.1.45	cleandomain	
7.1.40	fileversion	
7.1.48	createversion	
7.1.49	rollbackversion	
7.1.50	deleteversion	
7.1.51	deleteversions	
7.1.52	loadfile_header	
7.1.53	loadfile_fast	
7.1.54	loadfile	70
7.1.55	loadlockfile	70
7.1.56	savefile	70
7.1.57	savelockfile	71
7.1.58	appendfile	71
7.1.59	lockfile	
7.1.60	unlockfile	
7.1.61	deletefile	
7.1.62	restoremediafile	
7.1.63	preparemediafile	
7.1.64	deletemediafiles	
7.1.65	avoidfilecollision	
7.1.66	substr_in_array	
7.1.67	downloadobject	
7.1.68	downloadfile	
7.1.69	loadcontainer	
7.1.70	savecontainer	76
7.1.71	checkworkflow	76
7.1.72	inherit_db_load	77
7.1.73	inherit_db_read	77
7.1.74	inherit_db_close	77
7.1.75	inherit_db_save	
7.1.76	inherit_db_getparent	
7.1.77	inherit_db_getchild	
7.1.78	inherit_db_setparent	
7.1.79	inherit_db_insertparent	
7.1.79	inherit_db_deleteparent	
7.1.80	createinstance	
7.1.82	editinstance	
7.1.83	deleteinstance	
7.1.84	createpublication	
7.1.85	editpublication	
7.1.86	editpublicationsetting	
7.1.87	deletepublication	81
7.1.88	createpersonalization	82
7.1.89	deletepersonalization	82
7.1.90	createtemplate	83
7.1.91	loadtemplate	83
7.1.92	edittemplate	
7.1.93	deletetemplate	
7.1.94	createuser	
7.1.95	edituser	
7.1.96	deleteuser	
7.1.90		
	creategroup	
7.1.98	editgroupdeletegroup	
7.1.99	deletegroup	
7.1.100	renamegroupfolder	୪/

7.1.101	deletegroupfolder	
7.1.102	renameworkflowfolder	88
7.1.103	deleteworkflowfolder	
7.1.104	createmediacat	
7.1.105	renamemediacat	
7.1.106	deletemediacat	
7.1.107	uploadtomediacat	
7.1.108	deletefrommediacat	
7.1.109	createfolder	
7.1.110	createfolders	
7.1.111	collectfolders	
7.1.112	copyfolders	
7.1.113	deletefolder	
7.1.114	renamefolder	
7.1.115	correctcontainername	
7.1.116	createobject	
7.1.117	uploadfile	
7.1.118	createmediaobject	
7.1.119	createmediaobjects	
7.1.120	editmediaobject	
7.1.121 7.1.122	manipulateobject	
	deletemarkobject	
7.1.123 7.1.124	deleteunmarkobjectdeleteenhiest	
7.1.124 7.1.125	deleteobject	
7.1.125 7.1.126	renameobjectrenamefile	
7.1.120	cutobject	
7.1.127	copyobject	
7.1.128	copyconnectedobject	
7.1.127	pasteobject	
7.1.130	lockobject	
7.1.132	unlockobject	
7.1.133	publishobject	
7.1.134	publishlinkedobject	
7.1.135	unpublishobject	
7.1.136	processobjects	
7.1.137	collectobjects	
7.1.138	manipulateallobjects	
7.1.139	remoteclient	
7.1.140	HTTP_Post	
7.1.141	HTTP_Get	
7.1.142	HTTP_Proxy	
7.1.143	loadbalancer	
7.1.144	savelog	107
7.1.145	deletelogdeletelog	108
7.1.146	debuglog	108
7.1.147	notifyusers	
7.1.148	sendlicensenotification	109
7.1.149	licensenotification	109
7.1.150	html_diff	110
7.1.151	createfavorite	
7.1.152	deletefavorite	110
7.1.153	rewrite_targetURI	
7.1.154	rewrite_homepage	
7.1.155	sendmessage	
	API Functions	
	etserverload	
722 a	etconfigvalue	112

7.2.3	getsession	112
7.2.4	getrequest	112
7.2.5	getrequest_esc	113
7.2.6	getuserip	113
7.2.7	getobjectlistcells	
7.2.8	getlanguageoptions	
7.2.9	getlanguagefile	
7.2.10	getcodepage	
7.2.10	getcodepage	
7.2.12	getescapedtext	
7.2.13	getsearchhistory	
7.2.14	gettaxonomy_sublevel	
7.2.15	gettaxonomy_childs	
7.2.16	gethierarchy_defintion	
7.2.17	gethierarchy_sublevel	116
7.2.18	getkeywords	117
7.2.19	getmetakeywords	117
7.2.20	getmetadescription	117
7.2.21	getgooglesitemap	
7.2.22	getlistelements	
7.2.23	getmetadata	
7.2.24	getmetadata_multiobjects	
7.2.25	getmetadata_container	
7.2.26	getobjectcontainergetobjectcontainer	
7.2.27	• •	
— . — .	getcontainer	
7.2.28	getwallpaper	
7.2.29	getcontainername	
7.2.30	getlocationname	
7.2.31	getthemelocation	
7.2.32	getcategory	
7.2.33	getpublication	122
7.2.34	getlocation	122
7.2.35	getobject	122
7.2.36	getmediacontainername	122
7.2.37	getmediafileversion	123
7.2.38	getobjectid	
7.2.39	getobjectlink	
7.2.40	getcontainerversions	
7.2.41	getlocaltemplates	
7.2.42	gettemplates	
7.2.42	gettemplateversions	
7.2.43	· ·	
	getfileinfo	
7.2.45	getobjectinfo	
7.2.46	getfilesize	
7.2.47	getmimetype	
7.2.48	getfiletype	
7.2.49	getpreviewwidth	127
7.2.50	getpdfinfo	127
7.2.51	getvideoinfo	127
7.2.52	getbrowserinfo	128
7.2.53	getcontentlocation	
7.2.54	getmedialocation	
7.2.55	getlockedfileinfo	
7.2.56	getfavorites	
7.2.57	getboxesgetboxes	
7.2.58	getusersonline	
7.2.59	<u> </u>	
	getchatstate	
7.2.60	getimagelib	130

7.2.61	getfilenamegetfilename	130
7.2.62	gethypertag	
7.2.63	gethypertagname	
7.2.64	gethtmltag	
7.2.65	gethtmltags	
7.2.66	getattribute	
7.2.67	getoption	
7.2.68	getcharset	
7.2.69	getartid	
7.2.70	getelementid	
7.2.71	getfirstkey	
7.2.72	getdirectoryfiles	
7.2.72	getuserinformation	
7.2.73	getworkflowitem	
	et API Functions	
7.3.1	setsession	
7.3.2	settaxonomy	
7.3.3	setarticle	
7.3.4	settext	
7.3.5	setmedia	
7.3.6	setpagelink	
7.3.7	setcomplink	137
7.3.8	sethead	137
7.3.9	setfilename	138
7.3.10	setboxes	138
7.4 Cd	onnect API Functions	
7.4.1	ftp_userlogon	
7.4.2	ftp_userlogout	
7.4.3	ftp_getfile	
7.4.4	ftp_putfile	
7.4.4	ftp_filelist	
	<u>, —</u>	
7.4.6	createsharelink_facebook	
7.4.7	createsharelink_twitter	
7.4.8	createsharelink_googleplus	
7.4.9	createsharelink_linkedin	
7.4.10	createsharelink_pinterest	
7.5 S€	ecurity API Functions	
7.5.1	rootpermission	
7.5.2	globalpermission	142
7.5.3	localpermission	142
7.5.4	accessgeneral	142
7.5.5	accesspermission	143
7.5.6	setlocalpermission	143
7.5.7	checkpublicationpermission	
7.5.8	checkadminpermission	
7.5.9	checkrootpermission	
7.5.10	checkglobalpermission	
7.5.10	checklocalpermission	
	•	
7.5.12	userlogin	
7.5.13	registerinstance	
7.5.14	createchecksum	
7.5.15	writesession	
7.5.16	writesessiondata	
7.5.17	createsession	
7.5.18	killsession	147
7.5.19	checkdiskkey	
7.5.20	checkpassword	147
7.5.21	loguserip	148

7.5.22	checkuserip	
7.5.23	checkuserrequests	148
7.5.24	checkusersession	149
7.5.25	allowuserip	149
7.5.26	valid_objectnamevalid_objectname	149
7.5.27	valid_locationname	
7.5.28	valid_publicationname	
7.5.29	html_encode	
7.5.30	html_decode	
7.5.30		
	scriptcode_encode	
7.5.32	scriptcode_extract	
7.5.33	scriptcode_clean_functions	
7.5.34	sql_clean_functions	
7.5.35	url_encode	
7.5.36	url_decode	
7.5.37	shellcmd_encode	
7.5.38	hcms_crypt	153
7.5.39	hcms_encrypt	153
7.5.40	hcms_decrypt	153
7.5.41	createtimetoken	
7.5.42	checktimetoken	
7.5.43	createtoken	
7.5.44	checktoken	
7.5.45	createuniquetoken	
7.5.46	rand_secure	
	dia API Functions	
7.6.1	indexcontent	
7.6.2	unindexcontent	
7.6.3	reindexcontent	
7.6.4	createthumbnail_indesign	
7.6.5	createthumbnail_video	
7.6.6	createmedia	
7.6.7	convertmedia	158
7.6.8	convertimage	159
7.6.9	rotateimage	160
7.6.10	getimagecolors	160
7.6.11	getimagecolorkey	160
7.6.12	hex2rgb	
7.6.13	rgb2hex	
7.6.14	readmediaplayer_config	
7.6.15	savemediaplayer_config	
7.6.16	createdocument	
7.6.10	unzipfile	
	· ·	
7.6.18	clonefolder	
7.6.19	zipfiles	
7.6.20	px2mm	
7.6.21	px2inch	
7.6.22	inch2px	
7.6.23	mediasize2frame	165
7.6.24	vtt2array	165
7.7 Me	tadata API Functions	
7.7.1	importmetadata	
7.7.2	createtaxonomy	
7.7.2	splitkeywords	
7.7.4	copymetadata	
7.7.5	extractmetadata	
7.7.5 7.7.6		
7.7.6 7.7.7	xmlobject2arrayid3_getdata	
		16/

7	7.8	id3_writefile	160
	7.9	id3_create	
	7.10	xmp_getdata	
	7.11	xmp_writefile	
	7.12	xmp_create	
	7.13	geo2decimal	
7.	7.14	exif_getdata	
7.	7.15	iptc_getdata	
7.	7.16	iptc_getcharset	170
7.	7.17	iptc_maketag	171
7.	7.18	iptc_writefile	171
7.	7.19	iptc_create	171
7.	7.20	createmapping	172
7.	7.21	getmapping	
7.	7.22	setmetadata	
7.8	Lin	k API Functions	
	8.1	link_db_restore	
	8.2	link_db_load	
	8.3	link_db_read	
	8.4	link_db_close	
	8.5	link_db_save	
	8.6	link_db_update	
		·	
	8.7	link_db_insert	
	8.8	link_db_delete	
	8.9	link_db_getobject	
	8.10	link_update	
	8.11	getlinkedobject	
	8.12	getconnectedobject	
7.	8.13	extractlinks	
7.9		gin API Functions	178
7.	9.1	plugin_getdefaultconf	178
7.	9.2	plugin_readmenu	178
7.	9.3	plugin_parse	178
7.	9.4	plugin_generatedefinition	179
7.	9.5	plugin_saveconfig	179
7.	9.6	plugin_generatelink	
7.10) Use	er Interface API Functions	
	10.1	windowwidth	
7.	10.2	windowheight	
	10.3	toggleview	
	10.4	togglesidebar	
	10.5	setfilter	
	10.6	objectfilter	
	10.7	showshorttext	
	10.8	showtopbar	
	10.9	showtopmenubar	
	10.7	·	
	10.10	9	
		showinfopage	
	10.12		
	10.13		
	10.14		
	10.15	•	
	10.16		
	10.17	· ·	
	10.18		
	10.19		
	10.20	· —	
7.	10.21	showinlineeditor	188

	7.10.22	showvideoplayer	188
	7.10.23	showvideoplayer_head	189
	7.10.24	showaudioplayer	189
	7.10.25	showaudioplayer_head	189
	7.10.26	debug_getbacktracestring	190
	7.10.27	showAPIdocs	190
	7.10.28	readnavigation	190
	7.10.29	createnavigation	191
	7.10.30	shownavigation	
	7.10.31	showselect	
	7.10.32	showtranslator	
	7.10.33	showmapping	
7.	11 Temp	plate Engine API Functions	
	7.11.1	checklanguage	
	7.11.2	checkgroupaccess	
	7.11.3	transformlink	
	7.11.4	followlink	
	7.11.5	errorhandler	
	7.11.6	viewinclusions	
	7.11.7	buildview	
	7.11.8	buildsearchform	
	7.11.9	buildbarchart	
7		API Functions	
٠.	7.12.1	valid_tagname	
	7.12.1	setxmlparameter	
	7.12.2	getcontent	
	7.12.3		
	7.12.4	geticontentgetymleentent	
	7.12.5	getxmlcontent	
		getxmlicontent	
	7.12.7	selectcontent	
	7.12.8	selecticontent	
	7.12.9	selectxmlcontent	
	7.12.10	selectxmlicontent	
	7.12.11	deletecontent	
	7.12.12	deleteicontent	
	7.12.13	setcontent	
	7.12.14	seticontent	
	7.12.15	setcontent_fast	
	7.12.16	updatecontent	
	7.12.17	insertcontent	
	7.12.18	inserticontent	
	7.12.19	addcontent	
	7.12.20	addicontent	
7.	13 Repor	rt API Functions	
	7.13.1	createreport	206
	7.13.2	editreport	206
	7.13.3	loadreport	207
	7.13.4	deletereport	207
	7.13.5	analyzeSQLselect	207
	7.13.6	showpiechart	
	7.13.7	showcolumnchart	
	7.13.8	showtimelinechart	
	7.13.9	showgeolocationchart	
7.		ct API Functions	
	7.14.1	createproject	
	7.14.2	editproject	
	7.14.3	deleteproject	
	7.14.4	getprojectstructure	

	7.14	.5	showprojectrecordshowprojectrecord	212
	7.14	.6	showganttchart	212
	7.15	Task	API Functions	213
	7.15	.1	createtask	213
	7.15	.2	edittask	213
	7.15	.3	deletetask	214
	7.15	. 4	tasknotification	214
	7.16	Work	flow API Functions	215
	7.16	.1	deleteworkflow	215
	7.16	.2	buildworkflow	215
	7.16	.3	workflowaccept	215
	7.16	. 4	acceptobject	216
	7.16	.5	workflowreject	216
	7.16	.6	rejectobject	217
	7.17	Cloud	Storage API Functions	217
	7.17	.1	loadcloudobject	217
	7.17	.2	savecloudobject	217
	7.17	.3	copycloudobject	218
	7.17	. 4	renamecloudobject	218
	7.17	.5	deletecloudobject	219
	7.17	.6	synccloudobjects	219
	7.18	Impo	rt/Export API Functions	219
	7.18	.1	exportobjects	219
	7.18	.2	importobjects	220
8	Rech	itliche	Hinweise / Impressum	221
	8.1		en und Anregungen	
			essum	
	8.3	Recht	tliche Hinweise	221

1 Einleitung

Die folgenden Kapitel behandeln die Funktionsbibliotheken des hyper Content & Digital Asset Management Servers und stellen somit die Dokumentation des API (Application Programming Interface) dar.

Alle Bibliotheken befinden sich innerhalb der hyperCMS Installation im Ordner "function" und können in die jeweiligen Scripts bzw. Templates eingebunden und genutzt werden. Damit lassen sich z.B. auch dynamische Seiten (Applikationen) unter Einsatz des XML-Content-Repository programmieren.

Sollten Sie Ihre Applikation auf einen physisch getrennten Server betreiben, so ist es wichtig, dass die Funktionsbibliotheken auch auf dem Publikationsserver zur Verfügung stehen. In diesem Fall ist es wichtig, dass die entsprechenden Dateien auch am Publikationsserver zur Verfügung stehen.

2 hyperCMS XML-Content-Repository

Das XML-Content-Repository beinhaltet alle XML-Content-Container und stellt somit alle Inhalte native XML zur Verfügung. Die Struktur (Schema) innerhalb eines XML-Content-Containers wird auf Basis des verwendeten Templates dynamisch erzeugt und besitzt folgendes Aussehen:

```
<?xml version="1.0" encoding="UTF-8" ?>
<container>
 <hyperCMS>
  <contentcontainer>0000023.xml</contentcontainer>
  <contentxmlschema>object/page</contentxmlschema>
  <contentorigin>%page%/Publication/testpage.php</contentorigin>
  <contentobjects>%page%/Publication/testpage.php|%page%/ Publication/linkedcopy_of_testpage.php
|</contentobjects>
  <contentuser>demouser</contentuser>
  <contentcreated>2002-12-01 10:02:40</contentcreated>
  <contentdate>2004-11-26 14:32:33</contentdate>
  <contentpublished>2004-11-26 14:39:41</contentpublished>
 <contentstatus>active</contentstatus>
 </hyperCMS>
 <head>
  <pagetitle>test</pagetitle>
  <pageauthor>Mr. Content/pageauthor>
  <pagedescription>just a small demonstration</pagedescription>
  <pagekeywords>demo of XML</pagekeywords>
  <pagecontenttype>text/html; charset=UTF-8</pagecontenttype>
  <pagelanguage>de</pagelanguage>
  <pagerevisit></pagerevisit>
 </head>
 <textcollection>
  <text>
   <text_id>headline</text_id>
   <textuser>demouser</textuser>
   <textcontent>fqfdqfdq</textcontent>
  </text>
  <text>
   <text_id>summary</text_id>
   <textuser>demouser</textuser>
   <textcontent><![CDATA[This is a
   <STRONG><EM>summary</EM></STRONG>]]></textcontent>
  </text>
 </textcollection>
 <mediacollection>
  <media>
   <media_id>logo</media_id>
   <mediauser>otheruser</mediauser>
   <mediafile>Publication/demo_hcms0000033.jpg</mediafile>
   <mediaobject>%page%/Publication/Multimedia/demo.jpg</mediaobject>
   <mediaalttext>demoimage</mediaalttext>
```

```
<mediaalign></mediaalign>
    <mediawidth>200</mediawidth>
   <mediaheight>100</mediaheight>
  </media>
 </mediacollection>
 kcollection>
  k>
    link_id>verweis</link_id>
    <linkuser>demouser</linkuser>
    <linkhref>http://localhost/index.php</linkhref>
    <linktarget>_blank</linktarget>
    linktext>click me</linktext>
  </link>
 </linkcollection>
 <componentcollection>
  <component>
    <component_id>teasers/component_id>
    <componentuser>otheruser</componentuser>
    <componentcond>$customer == "private"</componentcond>
    <componentfiles>%comp%/Publication/teaser_1.php|%comp%/Publication/teaser_2.php|/componentfiles>
  </component>
  <component>
    <component_id>banner</component_id>
    <componentuser>demouser</componentuser>
    <componentcond></componentcond>
    <componentfiles>%comp%/banner.php</componentfiles>
  </component>
 </componentcollection>
 <articlecollection>
  <article>
    <article_id>news</article_id>
    <articletitle>Top News</articletitle>
    <articledatefrom>2002-10-01</articledatefrom>
    <articledateto>2002-11-01</articledateto>
    <articlestatus>active</articlestatus>
    <articleuser>demouser</articleuser>
    <articletextcollection>
     <text>
      <text id>news:headline</text id>
      <textuser>demouser</textuser>
      <textcontent>News from Scene</textcontent>
     </text>
    </articletextcollection>
    <articlemediacollection>
    </articlemediacollection>
    <articlelinkcollection>
    </articlelinkcollection>
    <articlecomponentcollection>
    </articlecomponentcollection>
  </article>
  <article>
    <article_id>special</article_id>
    <articletitle>Special Info</articletitle>
    <articledatefrom>2002-01-01</articledatefrom>
    <articledateto>2002-01-01</articledateto>
    <articlestatus>inactive</articlestatus>
    <articleuser>otheruser</articleuser>
    <articletextcollection>
     <text>
      <text_id>special:informations</text_id>
      <textuser>otheruser</textuser>
      <textcontent><![CDATA[<STRONG><FONT color=#cc0033>What is really going on behind the
Scene</FONT></STRONG>... find it out]]></textcontent>
     </text>
    </articletextcollection>
    <articlemediacollection>
    </articlemediacollection>
    <articlelinkcollection>
    </articlelinkcollection>
    <articlecomponentcollection>
    </articlecomponentcollection>
  </article>
 </articlecollection>
</container>
```

Nach Durchsicht des Content Containers ist eine Struktur zu erkennen, die sich aus den folgenden wesentlichen Grundelementen für die Content-Ablage zusammensetzt:

- hyperCMS spezifische Informationen
- Meta-Informationen
- Text
- Medien (Bilder oder andere Multimedia-Dateien)
- Links
- Komponenten
- Artikel

Der gesamte Inhalt setzt sich aus diesem Grundbausteinen zusammen, deren Informationen wiederum innerhalb von XML-Tags abgelegt werden.

Artikel nehmen so wiederum die Elemente Text, Medien und Links in sich auf. Der gesamte Inhalt einer Seite oder Komponente lässt sich über den zugehörigen Content-Container beziehen.

2.1 hyperCMS spezifische Informationen

Die in diesem XML-Knoten erfassten Daten stellen primär für das Management des Containers relevante Informationen dar.

<hvperCMS>

<contentcontainer>0000023.xml</contentcontainer>

- <contentxmlschema>object/page</contentxmlschema>
- <contentorigin>%page%/testpage.php</contentorigin>
- <contentobjects>%page%/testpage.php|%page%/linkedcopy_of_testpage.php |</contentobjects>
- <contentuser>demouser</contentuser>
- <contentdate>2002-11-26</contentdate>
- <contentpublished>2002-11-26</contentpublished>
- <contentstatus>active</contentstatus>
- </hyperCMS>

Erklärung:

contentcontainer contentxmlschema contentorigin

Name des Content Containers (einmalig über alle Publikationen) Schema des Objektes: Seite = page oder Komponente = comp Objekt (Seite oder Komponente) die zur Generierung des Content

Containers führte

contentobjects Alle Objekte die diesen Content Container benutzen

contentuser Objekteigentümer

contentdate Datum der letzten Änderung des Containers

contentpublished Datum der letzten Publizierung eines Objektes basierend auf den

Content Container

contentstatus Der Status ist "active" solange ein Objekt das auf den Container basiert

existiert. Wurden alle Objekte die auf den Container basieren entfernt wird der Status "deleted" gesetzt. Der Container beinhaltet damit den letzten Informationsstand, kann jedoch nicht mehr genutzt werden.

2.2 Meta-Informationen

Die Standard Meta-Informationen einer HTML-Seite werden in diesem XML-Knoten beschrieben.

```
<head>
  <pagetitle>test</pagetitle>
  <pageauthor>Mr. Content</pageauthor>
  <pagedescription>just a small demonstration</pagedescription>
  <pagekeywords>demo of XML</pagekeywords>
  <pagecontenttype>text/html; charset=UTF-8</pagecontenttype>
  <pagelanguage>de</pagelanguage>
  <pagerevisit></pagerevisit></head></pagerevisit></pagerevisit></pagerevisit></pagerevisit></pagerevisit></pagerevisit></pagerevisit></pagerevisit></pagerevisit></pagerevisit></pagerevisit></pagerevisit></pagerevisit></pagerevisit></pagerevisit></pagerevisit></pagerevisit></pagerevisit></pagerevisit></pagerevisit></pagerevisit></pagerevisit></pagerevisit></pagerevisit></pagerevisit></pagerevisit></pagerevisit></pagerevisit></pagerevisit></pagerevisit></pagerevisit></pagerevisit></pagerevisit></pagerevisit></pagerevisit></pagerevisit></pagerevisit></pagerevisit></pagerevisit></pagerevisit></pagerevisit></pagerevisit></pagerevisit></pagerevisit></pagerevisit></pagerevisit></pagerevisit></pagerevisit></pagerevisit></pagerevisit></pagerevisit></pagerevisit></pagerevisit></pagerevisit></pagerevisit></pagerevisit></pagerevisit></pagerevisit></pagerevisit></pagerevisit></pagerevisit></pagerevisit></pagerevisit></pagerevisit></pagerevisit></pagerevisit></pagerevisit></pagerevisit></pagerevisit></pagerevisit></pagerevisit></pagerevisit></pagerevisit></pagerevisit></pagerevisit></pagerevisit></pagerevisit></pagerevisit></pagerevisit></pagerevisit></pagerevisit></pagerevisit></pagerevisit></pagerevisit></pagerevisit></pagerevisit></pagerevisit></pagerevisit></pagerevisit></pagerevisit></pagerevisit></pagerevisit></pagerevisit></pagerevisit></pagerevisit></pagerevisit></pagerevisit></pagerevisit></pagerevisit></pagerevisit></pagerevisit></pagerevisit></pagerevisit></pagerevisit></pagerevisit></pagerevisit></pagerevisit></pagerevisit></pagerevisit></pagerevisit></pagerevisit></pagerevisit></pagerevisit></pagerevisit></pagerevisit></pagerevisit></pagerevisit></pagerevisit></pagerevisit></pagerevisit></pagerevisit></pagerevisit></pagerevisit></pa
```

Erklärung:

pagetitle Seitentitel pageauthor Seitenautor

pagedescription Beschreibung der Inhalte der Seite pagekeywords Liste der Schlüsselwörter der Seite

pagecontenttype Content-Type (Zeichensatz) der Seite oder Komponente

pagelanguage Sprachkürzel der Seite

pagerevisit Wiederbesuch der Seite durch Suchmaschinen

2.3 Text

Diese XML-Knoten speichern den Text.

```
<text>
<text_id>headline</text_id>
<textuser>demouser</textuser>
<textcontent>fgfdgfdg</textcontent>
</text>
```

Erklärung:

text_id Textidentifikation

textuser Texteigentümer (letzte Änderung des Textes durch einen Benutzer)

textcontent Inhalt des Textes

2.4 Medien

Dieser XML-Knoten beschreibt eingebunden Medien.

```
<media>
<media_id>logo</media_id>
<media_id>logo</media_id>
<mediauser>otheruser</mediauser>
<mediaile>Publication/demo_hcms0000033.jpg</mediafile>
<mediaobject>%page%/Publication/Multimedia/demo.jpg</mediaobject>
<mediaalttext>demoimage</mediaalttext>
<mediaalign></mediaalign>
<mediawidth>200</mediawidth>
<mediaheight>100</mediaheight>
</media>
```

Erklärung:

media_id Medienidentifikation

mediauser Medieneigentümer (letzte Änderung des Mediums durch einen Benutzer)

mediafile eingebunden Mediendatei mit Angabe der Publikation

mediaobject Pfadangabe zur Multimediakomponente

mediaalttext Alternativtext des Mediums mediaalign Ausrichtung des Mediums

medawidth Dargestellte Breite des Mediums mediaheight Dargestellte Höhe des Mediums

2.5 Links

Dieser XML-Knoten beschreibt die Verlinkung zu Seiten.

Erklärung:

link_id Linkidentifikation

linkuser Linkeigentümer (letzte Änderung des Links durch einen Benutzer)

linkhref Referenz (Link) zu einer Seite oder Datei linktarget Ziel des Referenzierung (Name des Frames)

linktext Text der den Link beschreibt/darstellt

2.6 Komponenten

Dieser XML-Knoten beschreibt die Verlinkung zu Komponenten.

```
<component>
  <component_id>teasers</component_id>
  <componentuser>otheruser</componentuser>
  <componentcond>$customer == "private"</componentcond>
  <componentfiles>%comp%/teaser_1.php|%comp%/teaser_2.php|</componentfiles>
</component>
```

Erklärung:

component_id Komponentenidentifikation

componentuser Komponenteneigentümer (letzte Änderung der Komponentenreferzierung

durch einen Benutzer)

componentcond Zugeordnetes Kundenprofil zu der Komponente

componentfiles Referenz (Komponenten-Link) zu einer oder mehreren Komponenten

2.7 Artikel

Dieser XML-Knoten beschreibt die Artikelinformation.

```
<article>
<article_id>news</article_id>
<article_id>news</article_id>
<articletitle>Top News</articletitle>
<articledatefrom>2002-10-01</articledatefrom>
<articledateto>2002-11-01</articledateto>
<articlestatus>active</articlestatus>
<articleuser>demouser</articleuser>
<articletextcollection>
</articletextcollection>
</article>
```

Erklärung:

article_id Artikelnidentifikation articletitle Titel des Artikels

articeldatefrom Beginn der Veröffentlichung des Artikels articeldateto Ende der Veröffentlichung des Artikels

articlestatus Bestimmung der Veröffentlichung des Artikels:

active = immer veröffentlicht inactive = nicht veröffenlicht

 $times witched \ = \ zeit gesteuerte \ Ver\"{o}ffentlichung$

articleuser Artikeleigentümer (letzte Änderung des Artikels durch einen Benutzer)

articlecollection Umfasst alle dem Artikel zugeordneten Inhalte

3 Funktionshibliotheken

3.1 Einbindung einer Bibliothek

Das Einbinden einer Konfiguratzion oder Bibliothek setzt voraus, dass man den absoluten oder relativen Pfad zur Bibliothek kennt. Durch Verwendung der Funktion "require" oder "require_once" und der Angabe des Pfades inklusive der einzubinden Datei werden die enthaltenen Funktionen der Bilbliothek eingebunden. Sobald die Bibliothek eingebunden ist, können deren Funktionen im Script genutzt werden.

Um die hyperCMS-Funktionen nutzen zu können, bedarf es der Einbindung der Datei "hypercms_api.inc.php". Diese Datei beinhaltet alle für die Programmierung benötigten Funktionen.

```
// absolute Angabe unter MS Windows
require_once ("C:/inetpub/wwwroot/hypercms/function/hypercms_api.inc.php");
// relative Angabe unter MS Windows oder auch UNIX-Derivaten
require_once ("function/hypercms_api.inc.php");
```

3.2 Laden der Konfiguration

3.2.1 Content Management Server

Um die Konfiguration von hyperCMS nutzen zu können muss die entsprechende Datei geladen warden. Diese beinhaltet alle wesentlichen Einstellungen des zu behandelnden Mandanten (Site).

Mit Hilfe der Identifikation einer Publikation, z.B. mit der Variable \$site kann die Konfiguration einer Publikation geladen werden. Die hyperCMS Hauptkonfigurationsatei befindet sich im Verzeichnis "hypercms/config" und trägt den Namen "config.inc.php". Die publikationsspezifischen Konfigurationsateien befinden sich im Verzeichnis "data/config". Deren Dateiname setzt sich aus dem Namen der Publikation sowie der Endung ".inc.php" zusammen, Bsp: site.inc.php.

```
// Einbinden der Hauptkonfigurationsatei (auf Pfadangabe ist zu achten):
require_once ("C:/inetpub/wwwroot/hypercms/config.inc.php");

// Einbinden Konfiguration einer Publikation
// Achtung: Bitte verwenden Sie valid_publicationname, um den Namen zu verifizieren, bevor
// Sie die Datei einbinden
if (valid_publicationname ($site))
{
    require_once ($mgmt_config['abs_path_data']."config/".$site.".conf.php");
}
```

Die Config-Dateien können geöffnet und gelesen werden. Jeder Parameter wird darin beschrieben und steht für die Nutzung in Programmen zur Verfügung. Bitte werfen Sie daher einen Blick in die Konfiguration, um mehr über die Parameter und deren Namen zu erfahren.

Es ist auch notwendig eine Sprache zu wählen. Hierfür dient die Variabel \$lang. \$lang beinhaltet das Sprachkürzel, welche in der Konfiguration "hypercms/config/config.inc.php" eingesehen werden können.

```
// Setzen der Spracheinstellung für Nachrichten von Funktionen, Deutsch (de) $lang = "de";
```

Da Sie die Funktionen des hyper API benutzen möchten, müssen Sie auch noch dieses einbinden.

```
// Einbinden der Funktionsbibliothek: require_once ($mgmt_config['abs_path_cms']."/function/hypercms_api.inc.php");
```

Nun können Sie die Funktionen des APIO nutzen, um z.B. einen Content Container einer bestimmtes Objektes über unterschiedliche Methoden zu laden:

```
// Laden der Seite
$pagedata = loadfile ("%page%/MyPublication/home/", "index.php");
// Content Container Name auslesen
$contentcontainer = filepointer ($pagedata, "content");
// Laden des veröffentlichten Content Container aus dem Content Repository
$containerdata = loadcontainer ($contentcontainer, "published", $user);
// Oder noch einfacher direkt über den Objektpfad
$containerdata = getobjectcontainer ("MyPublication", "%page%/MyPublication/home/",
"index.php", $user);
```

Funktionen laden die Konfiguration einer Publikatzion, sollte diese nicht verfügbar sein. Da viele Funktionen die Einstellungen einer Publikation benötigen, ist es ratsam die Konfiguration immer einzubinden.

3.2.2 Publication Server

Beachten Sie, dass die Konfiguration des Publication Servers (Publikationsziel) davon getrennt in einer INI-Datei abgelegt ist. Benötigen Sie die Publikationziel-Einstellungen, so müssen Sie die INI-Datei laden und parsen. Danach stehen Ihnen die Variablen in einem Array zur Verfügung.

Die INI-Datei des Publikationszieles befindet sich im externen Repository im Verzeichnis "repository/config". Der Name der Datei entspricht dem Namen der Publikation mit der Dateierweiterung ".ini".

```
// laden und parsen der INI-Datei mit hilfe von PHP
$publ_config = parse_ini_file ("C:/inetpub/wwwroot/repository/config/Mandant_1.ini");
// Zugreifen auf die Variablen des Publikationszieles
echo "Das ist der Document Root der Seiten der Publikation:".$publ_config[abs_publ_page];
```

3.3 Globale Variablen

Viele Funktionen nutzen globale Variablen die in der Konfiguration gespeichert sind und den Funktionen zur Verfügung stehen. Sie sollten daher bei der Wahl der Variablennamen in Ihren eigenen Scripts acht geben, dass Sie nicht die von hyperCMS genutzen globalen Variablen verwenden.

Die folgende Liste zeigt alle globalen Variablen von hyperCMS, die nicht in eigenen Scripts manipuliert/verändert werden dürfen:

```
$mgmt_config
$lang
$lang_name
$lang_shortcut
$lang_codepage
$lang_shortcut_default
```

Viele globale Variablen von hyperCMS sind für die Verwendung in hyperCMS-Scripts und PHP-Scripts nützlich, diese stehen nur dann zur Verfügung, wenn die entsprechende Konfiguration zuvor geladen wurde, oder eine hyperCMS-Script (wird nur während des Publikationsprozesses ausgeführt) in Verwendung ist. Da dies bei der Voransicht als auch beim Publizieren von Seiten und Komponenten passiert, können diese Variablen in hyperCMS-Scripts genutzt werden. Bei dynamischen Applikationen, die bei jedem Aufruf der Seite oder Komponente durch einen Besucher ausgeführt werden, muss die Konfiguration direkt im Template eingebunden werden, sofern Variablen von hyperCMS benötigt werden.

Content Management Server:

\$lang Sprachkürzel It. config.inc.php

\$mgmt_config['url_path_cms'] URL des hyperCMS Root Verzeichnis It. config.inc.php **mgmt_config[**'abs_path_cms'] absoluter Pfad zum hyperCMS Root Verzeichnis It.

config.inc.php

\$mgmt_config['url_path_page'**]** URL des Doc Roots der Publikation im Managementsystem

\$mgmt_config['abs_path_page'**]** absoluter Pfad zum Doc Roots der Publikation im

Managementsystem

\$mgmt_config['url_path_comp'**]** URL des Komponenten Root der Publikation im

Managementsystem

mgmt_config['abs_path_comp'] absoluter Pfad zum Komponenten Roots der Publikation im

Managementsystem

Publication Server:

hyperCMS-Scripts können die Variablen ohne weiteres zutun nutzen. Die Werte werden im Array \$publ_config gespeichert, sind aber auch optional auch ohne Array nutzbar. Wird das Script/Anwendung bei jedem publikationsseitigen Aufruf ausgeführt, so ist die Konfigurationsdatei gesondert zu laden.

\$publ_config['url_publ_page'**]** URL des Doc Roots der Publikation im Publikationssystem

\$publ_config['abs_publ_page'**]** absoluter Pfad zum Doc Roots der Publikation im

Publikationssystem

\$publ_config['url_publ_comp'] URL des Komponenten Roots der Publikation im

Publikationssystem

\$publ_config['abs_publ_comp'**]** absoluter Pfad zum Komponenten Roots der Publikation im

Publikationssystem

Optional (veraltet):

\$url_publ_page URL des Doc Roots der Publikation im Publikationssystem

\$abs_publ_page absoluter Pfad zum Doc Roots der Publikation im

Publikationssystem

\$url_publ_compURL des Komponenten Roots der Publikation im

Publikationssystem

\$abs_publ_comp absoluter Pfad zum Komponenten Roots der Publikation im

Publikationssystem

3.4 Vorlagenvariablen

Es gibt auch die Möglichkeit mit hyperCMS-eigenen Vorlagenvariablen zu arbeiten. Diese Variablen stellen eine Besonderheit dar, da sie nicht mit hyperCMS-Script in Verbindung stehen müssen. Sie sind vielmehr Platzhalter für den Wert einer Variable und können in Vorlagen beliebig eingesetzt werden.

Diese neutrale Form der Variablen sollte primär in Templates Verwendung finden, da damit ein technologieneutraler Einsatz stattfinden kann.

Achten Sie bitte auf die Kleinschreibung aller Variablen!

%container% steht für den Namen des Content Containers eines Objektes.

%template% steht für den Dateinamen der verwendeten Vorlage des Objektes.

%object% steht für den Namen des Objektes.

%date% beschreibt das aktuelle Datum im Format JJJJ-MM-TT.

Für die Einbindung von Mediendateien wird eine Pfadvariable benutzt. Diese Pfadvariable wird beim Publizieren der Seite oder Komponente durch die URL (Adresse) der Konfiguration des Publikationszieles ersetzt:

%media% steht für die Pfadangabe (URL) des Content Medien Repository. **%tplmedia%** steht für die Pfadangabe (URL) des Vorlagen Medien Repository.

Auch die Wurzelverzeischnisse der Seiten und Komponenten (publikationsseitig!) lassen sich abrufen:

%url_page% steht für die Pfadangabe (URL) des Seiten-Wurzelverzeichnisses.
 %abs_page% steht für die absolute Pfadangabe des Seiten-Wurzelverzeichnisses.
 %url_comp% steht für die Pfadangabe (URL) des Komponenten-Wurzelverzeichnisses.
 %abs_comp% steht für die absolute Pfadangabe des Komponenten -Wurzelverzeichnisses.

In Zusammenhang mit der Nutzung des hyperCMS APIs ist es oft ratsam, bei Pfadangaben die Platzhalter %page% und %comp% zu nutzen. Diese Pfadvariablen lassen sich nur managementseitig nutzen, sie stehen für die Pfade zu den Wurzelverzeischnissen von Seiten und Komponenten.

Zu beachten ist, dass die Variable immer gepaart mit dem Publikationsnamen das Wurzelverzeichnis bildet, z.B:

%page%/besttrade/ Wurzelverzeichnis der Seiten der Publikation "besttrade"

%page%/Publikationsname/ steht für die absolute Pfadangabe des Seiten-Wurzelverzeichnisses.

%comp%/Publikationsname/ steht für die absolute Pfadangabe des Komponenten - Wurzelverzeichnisses.

3.5 Bibliothek Object Operation

Diese Bibliothek beinhaltet alle Funktionen für die Manipulation von Objekten (Seiten, Komponenten oder Dateien). Bitte benutzen Sie ausschließlich diese Funktionen für den Zugriff auf Objekte, die das System verwaltet.

3.5.1 createfolder

Syntax:

createfolder (\$site, \$location, \$foldernew, \$user)

Beschreibung:

Erzeugt einen neuen Ordner.

Bsp:

\$result = createfolder ("besttrade", "%page%/besttrade/", "company", "brown");

Input-Parameter:

\$site Name der Publikation

\$location absoluter Pfad (Position des neuen Ordners)

\$foldernew Name des neuen Ordners

\$user Benutzername

globale Input-Parameter:

Die folgenden globalen Input Parameter sind ebenfalls der Funktion zu übergeben: \$lang Spracheinstellung bzw. Sprachkürzel, z.B. "en", "de"

Output:

Array \$result das folgende Informationen beinhaltet:

\$result[result] True/False (Konnte der neue Ordner angelegt werden)

\$result[add_onload] JavaScript Code für das onLoad Event

\$result[message] Nachricht über den Erfolg der Aktion bzw. Fehlermeldung

\$result[folder] Name des Ordners

3.5.2 deletefolder

Syntax:

deletefolder (\$site, \$location, \$folder, \$user)

Beschreibung:

Entfernt einen bestehenden Ordner. Der Ordner wird nur dann entfernt, wenn er keine Objekte mehr beinhaltet. Alle Objekte müssen daher zuvor mit deleteobject entfernt werden.

Bsp

\$result = deletefolder ("besttrade", "%page%/besttrade/", "company", "brown");

Input-Parameter:

\$site Name der Publikation

\$location absoluter Pfad (Position des neuen Ordners)

\$folder Name des zu entfernenden Ordners

\$user Benutzername

globale Input-Parameter:

Die folgenden globalen Input Parameter sind ebenfalls der Funktion zu übergeben: \$lang Spracheinstellung bzw. Sprachkürzel, z.B. "en", "de"

Output:

Array \$result das folgende Informationen beinhaltet:

\$result[result] True/False (Konnte der Ordner entfernt werden)

\$result[add_onload] JavaScript Code für das onLoad Event

\$result[message] Nachricht über den Erfolg der Aktion bzw. Fehlermeldung \$result[folder] Name des bestehenden Ordners bei Misserfolg, ansonst leer

3.5.3 renamefolder

Syntax:

renamefolder (\$site, \$location, \$folder, \$foldernew, \$user)

Beschreibung:

Benennt einen bestehenden Ordner um.

Bsp:

\$result = renamefolder ("besttrade", "%page%/besttrade/", "company", "news", "brown");

Input-Parameter:

\$site Name der Publikation

\$location absoluter Pfad (Position des Ordners)

\$folder Alter Name des Ordners \$foldernew Neuer Name des Ordners

\$user Benutzername

globale Input-Parameter:

Die folgenden globalen Input Parameter sind ebenfalls der Funktion zu übergeben: \$lang Spracheinstellung bzw. Sprachkürzel, z.B. "en", "de"

Output:

\$result[result] True/False (Konnte der Ordner umbenannt werden)

\$result[add_onload] JavaScript Code für das onLoad Event

\$result[message] Nachricht über den Erfolg der Aktion bzw. Fehlermeldung

\$result[folder] Name des Ordners

3.5.4 createobject

Syntax:

createobject (\$site, \$location, \$object, \$template, \$user)

Beschreibung:

Erzeugt eine neue Seite oder Komponente auf Basis einer Vorlage. Bitte beachten Sie das die Position (\$location) auch die Kategorie des Objektes (Seite/Komponente) bestimmt. Dies bedingt weiters, dass es sich beim Wert des Parameters \$template um eine gültige Seitenbzw. Komponentenvorlage handeln muss.

Bsp

\$result = createobject ("besttrade", "%page%/besttrade/", "index", "page_main", "Miller");

Input-Parameter:

\$site Name der Publikation

\$location absoluter Pfad (Position des Objektes)

\$object Name des neuen Objektes (Seite oder Komponente)

\$template Name der zu verwendenden Seiten- oder Komponentenvorlage

(Name der Vorlage oder Dateiname)

\$user Benutzername

globale Input-Parameter:

Die folgenden globalen Input Parameter sind ebenfalls der Funktion zu übergeben: \$lang Spracheinstellung bzw. Sprachkürzel, z.B. "en", "de"

Output:

Array \$result das folgende Informationen beinhaltet:

\$result[result] True/False (Erfolg der Aktion)

\$result[add_onload] JavaScript Code für das onLoad Event

\$result[message] Nachricht über den Erfolg der Aktion bzw. Fehlermeldung

3.5.5 deleteobject

Syntax:

deleteobject (\$site, \$location, \$object, \$user)

Beschreibung:

Entfernt eine bestehende Seite, Datei oder Komponente.

Bsp:

\$result = deleteobject ("besttrade", "%page%/besttrade/", "sales.doc", "Miller");

Input-Parameter:

\$site Name der Publikation

\$location absoluter Pfad (Position des Objektes)

\$object Name des Objektes \$user Benutzername

globale Input-Parameter:

Die folgenden globalen Input Parameter sind ebenfalls der Funktion zu übergeben: \$lang Spracheinstellung bzw. Sprachkürzel, z.B. "en", "de"

Output:

Array \$result das folgende Informationen beinhaltet:

\$result[result] True/False (Erfolg der Aktion)

\$result[add_onload] JavaScript Code für das onLoad Event

\$result[message] Nachricht über den Erfolg der Aktion bzw. Fehlermeldung

\$result[publication] Name der Publikation bzw. Mandant in dem das Objekt existiert

\$result[location] absoluter Pfad im Filesystem (Position des Objektes)

3.5.6 renameobject

Syntax:

renameobject (\$site, \$location, \$object, \$objectnew, \$user)

Beschreibung:

Umbenennen eine bestehender Seite, Datei oder Komponente.

Bsp

\$result = renameobject ("besttrade", "%page%/besttrade/", "sales.doc", "best.doc",
"Miller");

Input-Parameter:

\$site Name der Publikation

\$location absoluter Pfad (Position des Objektes)

\$object Alter Name des Objektes

\$objectnew Neuer Name des Objektes (ohne Dateiendung)

\$user Benutzername

globale Input-Parameter:

Die folgenden globalen Input Parameter sind ebenfalls der Funktion zu übergeben: \$lang Spracheinstellung bzw. Sprachkürzel, z.B. "en", "de"

Output:

Array \$result das folgende Informationen beinhaltet:

\$result[result] True/False (Erfolg der Aktion)

\$result[add_onload] JavaScript Code für das onLoad Event

\$result[message] Nachricht über den Erfolg der Aktion bzw. Fehlermeldung

\$result[publication] Name der Publikation bzw. Mandant in dem das Objekt existiert

\$result[location] absoluter Pfad im Filesystem (Position des Objektes)

3.5.7 cutobject

Syntax:

cutobject (\$site, \$location, \$object, \$user)

Beschreibung:

Ausschneiden eine bestehender Seite, Datei oder Komponente.

Bsp:

\$result = cutobject ("besttrade", "%page%/besttrade/", "index.php", "Miller");

Input-Parameter:

\$site Name der Publikation

\$location absoluter Pfad (Position des Objektes)

\$object Alter Name des Objektes

\$user Benutzername

globale Input-Parameter:

Die folgenden globalen Input Parameter sind ebenfalls der Funktion zu übergeben: \$lang Spracheinstellung bzw. Sprachkürzel, z.B. "en", "de"

Output:

Array \$result das folgende Informationen beinhaltet:

\$result[result] True/False (Erfolg der Aktion)

\$result[add_onload] JavaScript Code für das onLoad Event

\$result[message] Nachricht über den Erfolg der Aktion bzw. Fehlermeldung

\$result[object] Dateiname der Seite, Datei oder Komponente \$result[objectname] Name der Seite, Datei oder Komponente \$result[objecttype] Filetype bzw. File Extension der Datei

\$result[clipboard] temporärer Eintrag im Clipboard (kann als globale Variable

\$clipboard der Funktion pasteobject übergeben werden, diese muss somit keinen Lesezugriff auf die temporäre Datei ausführen)

3.5.8 copyobject

Syntax:

copyobject (\$site, \$location, \$object, \$user)

Beschreibung:

Kopieren eine bestehender Seite, Datei oder Komponente.

Bsp:

\$result = copyobject ("besttrade", "%page%/besttrade/", "index.php", "Miller");

Input-Parameter:

\$site Name der Publikation

\$location absoluter Pfad (Position des Objektes)

\$object Alter Name des Objektes

\$user Benutzername

globale Input-Parameter:

Die folgenden globalen Input Parameter sind ebenfalls der Funktion zu übergeben: \$lang Spracheinstellung bzw. Sprachkürzel, z.B. "en", "de"

Output:

Array \$result das folgende Informationen beinhaltet:

\$result[result] True/False (Erfolg der Aktion)

\$result[add_onload] JavaScript Code für das onLoad Event

\$result[message] Nachricht über den Erfolg der Aktion bzw. Fehlermeldung

\$result[object] Dateiname der Seite, Datei oder Komponente \$result[objectname] Name der Seite, Datei oder Komponente \$result[objecttype] Filetype bzw. File Extension der Datei

\$result[clipboard] temporärer Eintrag im Clipboard (kann als globale Variable

\$clipboard der Funktion pasteobject übergeben werden, diese muss somit keinen Lesezugriff auf die temporäre Datei ausführen)

3.5.9 copyconnectedobject

Syntax:

copyconnectedobject (\$site, \$location, \$object, \$user)

Beschreibung:

Kopieren eine bestehender Seite, Datei oder Komponente auf Basis des gleichen Content Containers.

Bsp:

\$result = copyconnectedobject ("besttrade", "%page%/besttrade/", "index.php", "Miller");

Input-Parameter:

\$site Name der Publikation

\$location absoluter Pfad (Position des Objektes)

\$object Alter Name des Objektes

\$user Benutzername

globale Input-Parameter:

Die folgenden globalen Input Parameter sind ebenfalls der Funktion zu übergeben: \$lang Spracheinstellung bzw. Sprachkürzel, z.B. "en", "de"

Output:

Array \$result das folgende Informationen beinhaltet:

\$result[result] True/False (Erfolg der Aktion)

\$result[add_onload] JavaScript Code für das onLoad Event

\$result[message] Nachricht über den Erfolg der Aktion bzw. Fehlermeldung

\$result[object] Dateiname der Seite, Datei oder Komponente \$result[objectname] Name der Seite, Datei oder Komponente \$result[objecttype] Filetype bzw. File Extension der Datei

\$result[clipboard] temporarer Eintrag im Clipboard (kann als globale Variable

\$clipboard der Funktion pasteobject übergeben werden, diese muss somit keinen Lesezugriff auf die temporäre Datei ausführen)

3.5.10 pasteobject

Syntax:

pasteobject (\$site, \$location, \$user)

Beschreibung:

Einfügen einer bestehender Seite, Datei oder Komponente.

Bsp:

\$result = pasteobject ("besttrade", "%page%/besttrade/", "Miller");

Input-Parameter:

\$site Name der Publikation

\$location absoluter Pfad (Position des Objektes)

\$user Benutzername

\$clipboard globale Varibale mit dem temporären Eintrag im Clipboard (Damit

ist ein Lesezugriff auf die temporäre Datei nicht notwendig.)

globale Input-Parameter:

Die folgenden globalen Input Parameter sind ebenfalls der Funktion zu übergeben: \$lang Spracheinstellung bzw. Sprachkürzel, z.B. "en", "de"

Output:

Array \$result das folgende Informationen beinhaltet:

\$result[result] True/False (Erfolg der Aktion)

\$result[add_onload] JavaScript Code für das onLoad Event

\$result[message] Nachricht über den Erfolg der Aktion bzw. Fehlermeldung

\$result[publication] Name der Publikation bzw. Mandant in dem das Objekt existiert

\$result[location] absoluter Pfad im Filesystem (Position des Objektes)

3.5.11 lockobject

Syntax:

lockobject (\$site, \$location, \$object, \$user)

Beschreibung:

Sperren einer oder mehrerer bestehender Seiten oder Komponenten die auf den gleichen Content Container beruhen für die exklusive Nutzung eines Benutzers.

Bsp

\$result = lockobject ("besttrade", "%page%/besttrade/", "index.php", "Miller");

Input-Parameter:

\$site Name der Publikation

\$location absoluter Pfad (Position des Objektes)

\$object Name des Objektes \$user Benutzername

globale Input-Parameter:

Die folgenden globalen Input Parameter sind ebenfalls der Funktion zu übergeben: \$lang Spracheinstellung bzw. Sprachkürzel, z.B. "en", "de"

Output:

Array \$result das folgende Informationen beinhaltet:

\$result[result] True/False (Erfolg der Aktion)

\$result[add_onload] JavaScript Code für das onLoad Event

\$result[message] Nachricht über den Erfolg der Aktion bzw. Fehlermeldung

3.5.12 unlockobject

Syntax:

unlockobject (\$site, \$location, \$object, \$user)

Beschreibung:

Entsperren einer oder mehrerer bestehender Seiten oder Komponenten die auf den gleichen Content Container beruhen für die gemeinsame Nutzung durch alle Benutzer.

Bsp

\$result = unlockobject ("besttrade", "%page%/besttrade/", "index.php", "Miller");

Input-Parameter:

\$site Name der Publikation

\$location absoluter Pfad (Position des Objektes)

\$object Name des Objektes \$user Benutzername

globale Input-Parameter:

Die folgenden globalen Input Parameter sind ebenfalls der Funktion zu übergeben: \$lang Spracheinstellung bzw. Sprachkürzel, z.B. "en", "de"

Output:

Array \$result das folgende Informationen beinhaltet:

\$result[result] True/False (Erfolg der Aktion)

\$result[add_onload] JavaScript Code für das onLoad Event

\$result[message] Nachricht über den Erfolg der Aktion bzw. Fehlermeldung

3.5.13 publishobject

Syntax:

publishobject (\$site, \$location, \$object, \$user)

Beschreibung:

Publizieren einer Seite oder Komponente. Alle gebundenen Kopien des Objektes bzw. dessen Content Containers werden ebenfalls publiziert. Gestattet die Berechtigung eines im Einsatz befindlichen Workflows die Publizierung nicht, so wird das Objekt auch nicht publiziert.

Bsp:

\$result = publishobject ("besttrade", "%page%/besttrade/", "index.php", "Miller");

Input-Parameter:

\$site Name der Publikation

\$location absoluter Pfad (Position des Objektes)

\$object Name des Objektes \$user Benutzername

globale Input-Parameter:

Die folgenden globalen Input Parameter sind ebenfalls der Funktion zu übergeben: \$lang Spracheinstellung bzw. Sprachkürzel, z.B. "en", "de"

Output:

Array \$result das folgende Informationen beinhaltet:

\$result[result] True/False (Erfolg der Aktion)

\$result[add_onload] JavaScript Code für das onLoad Event

\$result[message] Nachricht über den Erfolg der Aktion bzw. Fehlermeldung

3.5.14 unpublishobject

Syntax:

unpublishobject (\$site, \$location, \$object, \$user)

Beschreibung:

Entpublizieren einer Seite oder Komponente. Link und Task Management werden automatisch ausgeführt. Alle gebundenen Kopien des Objektes bzw. dessen Content Containers werden ebenfalls entpubliziert.

Bsp:

\$result = unpublishobject ("besttrade", "%page%/besttrade/", "index.php", "Miller");

Input-Parameter:

\$site Name der Publikation

\$location absoluter Pfad (Position des Objektes)

\$object Name des Objektes \$user Benutzername

globale Input-Parameter:

Die folgenden globalen Input Parameter sind ebenfalls der Funktion zu übergeben: \$lang Spracheinstellung bzw. Sprachkürzel, z.B. "en", "de"

Output:

Array \$result das folgende Informationen beinhaltet:

\$result[result] True/False (Erfolg der Aktion)

\$result[add_onload] JavaScript Code für das onLoad Event

\$result[message] Nachricht über den Erfolg der Aktion bzw. Fehlermeldung

3.5.15 getlinkedobject

Syntax:

getlinkedobject (\$site, \$location, \$object, \$cat)

Beschreibung:

Diese Funktion extrahiert alle Objekte, die auf das gegebene Objekt zeigen. Dies können Seiten-Links oder auch Komponenten-Links sein. Ist das übergebene Objekt eine Seite, so werden alle Objekte ermittelt die einen Seiten-Link auf das Objekt besitzen. Ist das übergebene Objekt eine Komponente, so werden alle Objekte gefunden die einen Komponenten-Link zu dem Objekt besitzen.

Bsp:

\$result = getlinkedobject ("besttrade", "%page%/besttrade/", "index.php", "page");

Input-Parameter:

\$site Name der Publikation

\$location absoluter Pfad (Position des Objektes)

\$object Name des Objektes

\$cat optional: Objekt Kategorie [page, comp]

Output:

Array \$result das folgende Informationen beinhaltet:

\$result False (Aktion fehlgeschlagen)

\$result[publication] Name der Publikation bzw. Mandant in dem das Objekt existiert

\$result[location] absoluter Pfad im Filesystem (Position des Objektes)

\$result[object] Name des Objektes

\$result[category] Kategorie des Objektes [page, comp]

3.5.16 getconnectedobject

Syntax:

getconnectedobject (\$site, \$container)

Beschreibung:

Diese Funktion ermittelt alle Objekte, die auf dem gleichen Content Container basieren. Der Name des Content Containers eines Objektes kann mittels der Funktion "getfilename" ermittelt werden.

Bsp:

\$result = getconnectedobject ("besttrade", "0000127.xml");

Input-Parameter:

\$site Name der Publikation

\$container Name des Content Containers

Output:

Array \$result das folgende Informationen beinhaltet:

\$result False (Aktion fehlgeschlagen)

\$result[publication] Name der Publikation bzw. Mandant in dem das Objekt existiert

\$result[location] absoluter Pfad im Filesystem (Position des Objektes)

\$result[object] Name des Objektes

\$result[category] Kategorie des Objektes [page, comp]

3.5.17 getobjectcontainer

Syntax:

getobjectcontainer (\$site, \$location, \$object, \$user)

Beschreibung:

Diese Funktion ladet den Content Container (XML-String) eines bestimmten Objektes. Das Objekt kann eine Seite, Datei, Komponente oder ein Ordner sein.

Die gewünschten Daten können mittels der Funktionen "getcontent" oder "selectcontent" aus dem XML-String ermittelt werden.

Bsp

\$xmldata = getobjectcontainer ("besttrade", "%page%/Home/", "index.php", "demouser");

Input-Parameter:

\$site Name der Publikation

\$location Pfad im Filesystem (Position des Objektes)

\$object Name des Objektes \$user Benutzername

Output:

XML-String Rückgabe des geladenen Content Containers

False Fehler aufgetreten

3.5.18 loadcontainer

Syntax:

loadcontainer (\$container)

Beschreibung:

Diese Funktion ladet den Content Container (XML-String) anhand seines Namens oder anhand seiner ID (hier wird der aktuelle in Arbeit befindliche Container als Standard angenommen und geladen).

Die gewünschten Daten können mittels der Funktionen getcontent oder selectcontent aus dem XML-String ermittelt werden.

Bsp:

// publizierter Zustand

\$xmldata1 = loadcontainer ("00012345.xml");

// Zustand "in Arbeit"

\$xmldata2 = loadcontainer ("00012345");

Input-Parameter:

\$container Name oder ID (mit vorangestellten Nullen) des Containers

Output:

XML-String Rückgabe des geladenen Content Containers

3.5.19 savecontainer

Syntax:

savecontainer (\$container, \$xmldata)

Beschreibung:

Diese Funktion speichert den Content Container (XML-String) anhand seines Namens oder anhand seiner ID (hier wird der aktuelle in Arbeit befindliche Container als Standard angenommen und geladen).

Bsp:

// publizierter Zustand
\$result = savecontainer ("00012345.xml", \$xmldata);
// Zustand "in Arbeit"
\$result = savecontainer ("00012345", \$xmldata);

Input-Parameter:

\$container Name oder ID (mit vorangestellten Nullen) des Containers

\$xmldata Daten des Content Containers als XML-String

Output:

True Funktion wurde fehlerfrei ausgeführt

3.6 Bibliothek File Pointer

Diese Funktionsbibliothek ermöglicht Ihnen den XML-Content-Container aus einem Objekt zu bestimmen bzw. diesen auch neu zu setzen. Sie müssen hierfür das Objekt (Seite oder Komponente) zuvor geladen haben, z.B. via http oder über das Filesystem mit Hilfe von loadfile.

3.6.1 getfilename

Syntax:

getfilename (\$filedata, \$tagname)

Beschreibung:

Der Funktion wird der Inhalt einer Seite, eines Datei-Objektes oder Komponente und der gewünschte Tag-Name – entweder content, template oder media - übergeben. Daraufhin wird der Dateiname des zugeordneten Content Container, Templates oder der Multimedia Datei zurückgegeben.

Bsp:

// lade eine Seite

\$filedata = loadfile ("%page%/myPublication/home/", "index.php");

// Content Container auslesen

\$contentcontainer = filepointer (\$filedata, "content");

Input-Parameter:

\$filedata Content

\$tagname Tagname [content, template, media]

Output:

Filename Funktion wurde fehlerfrei ausgeführt und liefert Dateiname des Content

Containers oder Templates

False Fehler aufgetreten oder das Objekt wird nicht vom System verwaltet

3.6.2 setfilename

Syntax:

setfilename (\$filedata, \$tagname, \$value)

Beschreibung:

Der Funktion wird der Inhalt einer Seite, eines Datei-Objektes oder Komponente, der Tagname und ein Wert für den File-Parameter des Tags übergeben. Nach Abarbeitung der Funktion und Überprüfung wird True oder False zurückgegeben.

Bsp:

\$result = setfilename (\$filedata, "template", "fullpage.page.tpl");

Input-Parameter:

\$filedata Seiten-, Datei-Objekt- oder Komponenteninhalt

\$tagname Tagname [content, template, media]

\$value (Datei)name des Content Containers, der Multimedia-Datei oder Templates

Output:

True Funktion wurde fehlerfrei ausgeführt

3.7 Bibliothek File Operation

Die folgenden Funktionen für File-Operationen sollten keinesfalls benutzt werden, um Objekte (Seiten, Komponenten oder Dateien) zu laden oder zu speichern.

Sie können diese Funktionen jedoch zum Laden und Speichern von XML-Content-Container verwenden, sollten Sie dies für die Entwicklung von Erweiterungen oder Anwendungen benötigen.

3.7.1 loadfile

Syntax:

loadfile (\$abs_path, \$filename)

Beschreibung:

Mit hilde dieser Funktion können Dateien geladen werden. Es müssen der Absolutpfad, als auch der Dateiname selbst als Parameter übergeben werden. Die Funktion wartet üblicherweise bis zu 3 Sekunden lang beim Laden von gesperrter Dateien. Wird der User-Parameter \$user gesetzt, so kann die Funktion auch gesperrte Dateien des gegeben Benutzers lesen.

Bsp:

\$data = loadfile ("%page%/myPublication/home/", "index.php");

Input-Parameter:

\$abs_path absoluter Pfad zur gewünschten Datei, %page% und %comp% können in der

Pfadangabe verwendet werden

\$filename Dateiname

Output:

Dateinhalt Funktion wurde fehlerfrei ausgeführt und liefert den Inhalt der Datei

False Fehler aufgetreten

3.7.2 savefile

Syntax:

savefile (\$abs_path, \$filename, \$filedata)

Beschreibung:

Mit savefile werden Dateien gespeichert. Hierfür müssen der Absolutpfad, der gewünschte Dateiname, als auch der Inhalt, die in die Datei geschrieben werden soll als Parameter übergeben werden. Ist die Datei gesperrt, so wird nicht gespeichert und False retourniert.

Bsp

\$result = savefile ("%page%/myPublication/home/", "index.php", "file content");

Input-Parameter:

\$abs_path absoluter Pfad zur gewünschten Datei, %page% und %comp% können in der

Pfadangabe verwendet werden

\$filename Dateiname

\$filedata Inhalt, der in die Datei geschrieben werden soll

Output:

True Funktion wurde fehlerfrei ausgeführt

3.7.3 loadlockfile

Syntax:

loadlockfile (\$user, \$abs_path, \$filename)

Beschreibung:

Damit können Dateien wie mit loadfile geladen werden, es wird aber zusätzlich ein Sperr-Mechanismus ausgelöst.

Diese Funktion sollte nur dann genutzt werden, wenn Daten manipuliert und wieder gespeichert werden sollen. Damit wird sicher gestellt, dass keine anderen Schreibzugriffe eines anderen Users erfolgen können. Beim Speichern muss die Funktion "savelockfile" benutzt werden, um den Inhalt wieder freizugeben.

Es müssen der Benutzer, der absolute Pfad als auch der Dateiname selbst als Parameter übergeben werden.

Bsp

\$data = loadlockfile ("Miller", "%page%/myPublication/home/", "index.php");

Input-Parameter:

\$user Benutzer der die Datei sperrt

\$abs_path absoluter Pfad zur gewünschten Datei, %page% und %comp% können in der

Pfadangabe verwendet werden

\$filename Dateiname

Output:

Dateinhalt Funktion wurde fehlerfrei ausgeführt und liefert den Inhalt der Datei

False Fehler aufgetreten

3.7.4 savelockfile

Syntax:

savelockfile (\$user, \$abs_path, \$filename, \$filedata)

Beschreibung:

Mit savelockfile werden Dateien gespeichert und entsperrt, die vorher mit loadlockfile geöffnet wurden. Hierfür müssen der Benutzer, der Absolutpfad, der gewünschte Dateiname, als auch der Inhalt, die in die Datei geschrieben werden soll als Parameter übergeben werden.

Bsp:

savelockfile ("Miller", "%page%/myPublication/home/", "index.php", "file content");

Input-Parameter:

\$user Benutzer der die Datei sperrt

\$abs_path absoluter Pfad zur gewünschten Datei, %page% und %comp% können in der

Pfadangabe verwendet werden

\$filename Dateiname

\$filedata Inhalt, der in die Datei geschrieben werden soll

Output:

True Funktion wurde fehlerfrei ausgeführt

3.7.5 lockfile

Syntax:

lockfile (\$user, \$abs_path, \$filename)

Beschreibung:

Mit lockfile werden Dateien von einem bestimmten Benutzer gesperrt und stehen für dessen exklusive Nutzung zur Verfügung. Hierfür müssen der Benutzer, der Absolutpfad, der gewünschte Dateiname als Parameter übergeben werden.

Bsp:

lockfile ("Miller", "%page%/myPublication/home/", "index.php");

Input-Parameter:

\$user Benutzer der die Datei sperrt

\$abs_path absoluter Pfad zur gewünschten Datei, %page% und %comp% können in der

Pfadangabe verwendet werden

\$filename Dateiname

Output:

True Funktion wurde fehlerfrei ausgeführt

False Fehler aufgetreten

3.7.6 unlockfile

Syntax:

unlockfile (\$user, \$abs_path, \$filename)

Beschreibung:

Mit unlockfile werden Dateien entsperrt, die vorher mit lockfile gesperrt oder mit loadlockfile geöffnet wurden. Hierfür müssen der Benutzer, der Absolutpfad, der gewünschte Dateiname als Parameter übergeben werden.

Bsp:

unlockfile ("Miller", "%page%/myPublication/home/", "index.php");

Input-Parameter:

\$user Benutzer der die Datei sperrt

\$abs_path absoluter Pfad zur gewünschten Datei, %page% und %comp% können in der

Pfadangabe verwendet werden

\$filename Dateiname

Output:

True Funktion wurde fehlerfrei ausgeführt

3.7.7 deletefile

Syntax:

deletefile (\$location, \$file, \$recursive)

Beschreibung:

Mit deletefile können Dateien und (leere) Ordner gelöscht werden. Es wird der Pfad der gewünschten Datei übergeben, der Dateiname, und ein Parameter "Rekursiv", der entweder (0) oder (1) beträgt. Wenn recursive 1 gesetzt wurde, wird der gesamte Inhalt des Ordners behandelt, also auch Unterverzeichnisse und deren Dateien, bei 0 werden nur die Dateien des angesprochenen Ordners (falls leer) entfernt.

Bsp:

deletefile ("%page%/myPublication/home/", "index.php", 0);

Input-Parameter:

\$abs_path absoluter Pfad zur gewünschten Datei, %page% und %comp% können in der

Pfadangabe verwendet werden

\$file Dateiname

\$recursive 0 oder 1, je nachdem ob sich der Vorgang auch auf Unterverzeichnisse

auswirken soll

Output:

True Funktion wurde fehlerfrei ausgeführt

False Fehler aufgetreten

3.7.8 appendfile

Syntax:

append (\$abs_path, \$filename, \$filedata)

Beschreibung:

Mit appendfile können Inhalte an Dateien angefügt werden. Die Funktion arbeitet wie savefile, der Unterschied besteht allerdings darin, dass bereits vorhandene Daten nicht überschrieben, sondern ergänzt werden. Hierfür müssen der Absolutpfad, der gewünschte Dateiname, als auch der Inhalt, die in die Datei geschrieben werden soll als Parameter übergeben werden.

Bsp:

appendfile ("%page%/myPublication/home/", "index.php", "© 2003 ...");

Input-Parameter:

\$abs_path absoluter Pfad zur gewünschten Datei, %page% und %comp% können in der

Pfadangabe verwendet werden

\$filename Dateiname

\$filedata Inhalt, der an die Datei angefügt werden soll

Output:

True Funktion wurde fehlerfrei ausgeführt

3.8 Bibliothek Edit Content

Die folgenden Funktionen bieten Ihnen die Möglichkeit Inhalte aus XML-Content-Container zu lesen und zu schreiben. Sie können optional auch mit anderen Technologien, die mit XML umgehen können, die Inhalte der Container abfragen. Die Bibliothek Edit Content bietet Ihnen jedoch eine sehr einfache als auch performante Methode hierfür.

3.8.1 setxmlparameter

Syntax:

setxmlparameter (\$xmldata, \$parameter, \$value)

Beschreibung:

Setzt den Wert eines bestimmten Parameters innerhalb der XML-Deklaration (1.Zeile).

Bsp

\$xmldata = setxmlparameter (\$xmldata, "encoding", "UTF-8");

Input-Parameter:

\$xmldata XML-String der übergeben und manipuliert werden soll \$parameter Name des Parameter dessen Wert geändert werden soll

\$value Wert des Paramaters

Output:

XML-String Rückgabe des manipulierten XML-Strings

3.8.2 getcontent

Syntax:

getcontent (\$xmldata, \$tag)

Beschreibung:

Holt den XML-Content aus dem Content Container, der sich innerhalb der Tags \$tag befindet. Ein Array mit allen gefundenen Inhalten bzw. Childs wird zurückgegeben und kann in einer Array-Variable gespeichert und weiterverwendet werden.

Bsp:

// hole alle text-childs aus Content Container
\$text_array = getcontent (\$xmldata, "<text>");

// ausgeben aller Text-Childs
foreach (\$text_array as \$text) echo \$text;

Input-Parameter:

\$xmldata XML-String der übergeben und durchsucht werden soll \$tag XML-Tag der die Information bzw. Childs umschliesst

Output:

Array Array mit allen gefunden Inhalten, der erste Wert/Inhalt kann mit Array[0]

angesprochen werden

3.8.3 getxmlcontent

Syntax:

getxmlcontent (\$xmldata, \$tag)

Beschreibung:

Holt den XML-Content aus dem Content Container, der sich innerhalb der Tags \$tag befindet und belässt im Unterschied zu getcontent die Tags im Rückgabewert (Array). Ein gesamter Node (well-formed) wird daher zurückgeliefert.

Ein Array mit allen gefundenen Inhalten bzw. Childs wird zurückgegeben und kann in einer Variable vom Typ Array gespeichert und weiterverwendet werden.

Bsp:

\$text_array = getxmlcontent (\$xmldata, "<text>");
foreach (\$text_array as \$text) echo \$text;

Input-Parameter:

\$xmldata XML-String der übergeben und durchsucht werden soll \$tag XML-Tag der die Information bzw. Childs umschliesst

Output:

Array Array mit allen gefunden Inhalten, der erste Wert/Inhalt kann mit Array[0]

angesprochen werden

3.8.4 selectcontent

Syntax:

selectcontent (\$xmldata, \$parenttag, \$childtag, \$childvalue)

Beschreibung:

Holt jenen XML-Content bestimmt durch \$parenttag aus dem Content-Container, der innerhalb des Childtags \$childtag einen bestimmten Wert \$childvalue aufweist. Ein Array mit allen gefundenen Inhalten wird zurückgegeben und kann in einer Array-Variable gespeichert und weiterverwendet werden.

Bsp:

```
Auszug aus dem Content Container:
```

```
<text>
    <text_id>summary</text_id>
    <textuser>editor1</textuser>
    <textcontent>This is my summary!</textcontent>
</text>

// hole alle Text-Childs mit der id=summary
$text_array = selectcontent ($xmldata, "<text>", "<text_id>", "summary");

// extrahiere das Summary aus dem gefundenen Inhalt
foreach ($text_array as $text)
{
    $summary = getcontent ($text, "<textcontent>");
}
```

Input-Parameter:

\$xmldata XML-String der übergeben und durchsucht werden soll \$parenttag XML-Tag der die Information bzw. das Child beinhaltet

\$childtag optional: XML-Tag der die Information umschliesst, die einen gewissen Wert

besitzen muss

\$childvalue optional: Wert der Bedingung, das WildCard Zeichen * kann am Anfang

und/oder am Ende des Ausdruckes verwendet werden und ist Platzhalter für

beliebige weitere Zeichen.

Output:

Array Array mit allen gefunden Inhalten, der erste Wert/Inhalt kann mit Array[0]

angesprochen werden

3.8.5 selectxmlcontent

Syntax:

selectxmlcontent (\$xmldata, \$parenttag, \$childtag, \$childvalue)

Beschreibung:

Holt jenen XML-Content definiert durch \$parenttag aus dem Content-Container, der innerhalb eines Childtags \$childtag einen bestimmten Wert \$childvalue aufweist. Im Unterschied zu getcontent werden die Parent-Tags im Rückgabewert (Array) belassen.

Ein Array mit allen gefundenen Inhalten wird zurückgegeben und kann in einer Array-Variable gespeichert und weiterverwendet werden.

```
Bsp:
```

```
Auszug aus dem Content Container:
.....
<text>
    <text_id>summary</text_id>
    <textuser>editor1</textuser>
    <textcontent>This is my summary!</textcontent>
</text>
.....

// hole alle Text-Childs mit der id=summary
$text_array = selectxmlcontent ($xmldata, "<text>", "<text_id>", "summary");

// extrahiere das Summary aus dem gefundenen Inhalt
foreach ($text_array as $text)
{
    $summary = getcontent ($text, "<textcontent>");
}
```

Input-Parameter:

\$xmldata XML-String der übergeben und durchsucht werden soll \$parenttag XML-Tag der die Information bzw. das Child beinhaltet

\$childtag optional: XML-Tag der die Information umschliesst, die einen gewissen Wert

besitzen muss

\$childvalue optional: Wert der Bedingung, das WildCard Zeichen * kann am Anfang

und/oder am Ende des Ausdruckes verwendet werden und ist Platzhalter für

beliebige weitere Zeichen.

Output:

Array Mrray mit allen gefunden Inhalten, der erste Wert/Inhalt kann mit Array[0]

angesprochen werden

3.8.6 deletecontent

Syntax:

deletecontent (\$xmldata, \$tagname, \$condtag, \$condvalue)

Beschreibung:

Löscht den gesamten XML-Content definiert durch den Tag \$tagname. Als Kriterium für die Auswahl der zu löschenden Childs wird das entsprechende XML-Childtag \$condtag und die umschlossene Information \$condvalue als Bedingung mitgeschickt.

Bsp:

Auszug aus dem Content Container:

```
<text>
<text>
<text_id>bedingung</text_id>
<textuser>editor1</textuser>
<textcontent>This is my summary!</textcontent>
</text>
.....
```

\$xmldata = deletecontent (\$xmldata, "<text>", "<text_id>", "bedingung");

Input-Parameter:

\$xmldata XML-String der übergeben wird

\$parenttag XML-Tag der die Information bzw. Childs umschliesst, die aus dem

Content Container entfernt werden sollen

\$condtag optional: Name des Parameters (XML-Child) das der Bedingung

unterliegt

\$condvalue optional: Wert der Bedingung, die erfüllt werden muss

Output:

XML-String Rückgabe des manipulierten XML-Strings

3.8.7 setcontent

Syntax:

setcontent (\$xmldata, \$parenttagname, \$tagname, \$contentnew, \$condtag, \$condvalue)

Beschreibung:

Ein XML-String wird übergeben und innerhalb eines bestimmten Parent Nodes (\$parenttagname) wird überprüft, ob ein bestimmter Parameter (\$condtag) existiert und einen bestimmter Wert (\$condvalue) aufweist. Ist die Bedingung erfüllt, wird der Wert des Parameters \$tagname durch einen neuen Wert \$contentnew ersetzt.

Bsp

Auszug aus dem Content Container:

```
<text>
  <text_id>bedingung</text_id>
  <textuser>editor1</textuser>
  <textcontent>This is should set!<textcontent>
</text>
.....
```

\$xmldata = setcontent (\$xmldata, "<text>", "<textcontent>", "This is my new value!",
"<text_id>", "bedingung");

Input-Parameter:

\$xmldata XML-String der übergeben und modifiziert werden soll

\$parenttagname optional: XML-Parenttag

\$tagname optional: XML-Childtag, dessen Wert ersetzt werden soll (wenn

Bedingung erfüllt)

\$contentnew Neuer Wert für den XML-Childtag \$tagmame

\$condtag optional: Name des Parameters der die Bedingung erfüllen muss

\$condvalue optional: Wert des Parameters für die Bedingung

Output:

XML-String Rückgabe des manipulierten XML-Strings

3.8.8 updatecontent

Syntax:

updatecontent (\$xmldata, \$xmlnode, \$xmlnodenew)

Beschreibung:

Alle XML-String \$xmlnode wird durch einen neuen String \$xmlnodenew in \$xmldata ersetzt. Diese Methode ist schneller als setcontent, wenn der aktualisierende XML Node bereits aus dem Container extrahiert wurde.

Bsp:

Auszug aus dem Content Container:

```
<text>
<text>
<text_id>bedingung</text_id>
<textuser>editor1</textuser>
<textcontent>This is old content!<textcontent>
</text>
```

\$xmldata = updatecontent (\$xmldata, "<textcontent>This is old content!<textcontent> ",
"<textcontent>This is my new content!<textcontent>");

Input-Parameter:

\$xmldata XML-String der übergeben und modifiziert werden soll

\$xmlnode zu ersetzender XML-String (Node bzw. Substring von \$xmldata) \$xmlnodenew optional: neuer XML-String, wenn leer, so wird der bestehende XML-

String entfernt.

Output:

XML-String Rückgabe des manipulierten XML-Strings

3.8.9 insertcontent

Syntax:

insertcontent (\$xmldata, \$insertxmldata, \$tagname)

Beschreibung:

Fügt einen XML-String (Child Node) vor dem Ende des übergebenen XML-Parenttags ein. Der modifizierte XML-String wird zurückgegeben.

Bsp:

```
Auszug aus dem Content Container:
```

\$xmldata = insertcontent (\$xmldata, \$insertxmldata, "<articletextlist>");

Input-Parameter:

\$xmldata XML-String der übergeben und modifiziert werden soll

\$insertxmldata XML-String der eingesetzt wird

\$tagname optional: XML-Parenttag an dessen Ende eingesetzt werden soll

Output:

XML-String Rückgabe des manipulierten XML-Strings

3.8.10 addcontent

Syntax:

addcontent (\$xmldata, \$sub_xmldata, \$grandtagname, \$condtag, \$condvalue, \$parenttagname, \$tagname, \$contentnew)

Beschreibung:

Innerhalb eines Parent Nodes wird ein Child Node hinzugefügt, sofern ein Wert im darüberliegenden Grandparent Node die Bedingung erfüllt. Im Child Node kann auf Wunsch gleichzeitig ein Wert gesetzt werden. Der modifizierte XML-String wird zurückgegeben.

```
Auszug aus dem Content Container:
<article>
 <article_id>art1</article_id>
 <articletitle></articletitle>
 <articledatefrom></articledatefrom>
 <articledateto></articledateto>
 <articlestatus>active</articlestatus>
 <articleuser></articleuser>
 <articletextlist>
  <text>
    <text id>art1:summary</text id>
    <textuser>editor1</textuser>
    <textcontent>This is my summary!</textcontent>
  </text>
----- hier wird ein Child Node eingefügt -----
  <text>
   <text_id>art1:longtext</text_id>
   <textuser>editor1</textuser>
   <textcontent>This is my summary!</textcontent>
  </text>
 </articletextlist>
</article>
$xmldata = addcontent ($xmldata, $sub_xmldata, "<article>", "<article_id>", "art1",
"<articletextlist>", "<text_id>", "art1:longtext");
Input-Parameter:
$xmldata
                          XML-String der übergeben und modifiziert werden soll
                          XML-String der eingebettet werden soll
$sub_xmldata
                          Enthält den XML-Childtag, in dem $sub_xmldata eingebettet
$grandtagname
                          werden soll
$condtag
                          optional: Name des Parameters der überprüft werden soll
$condvalue
                          optional: Wert des Parameters der überprüft werden soll
```

optional: XML-Childtag, in dem \$sub_xmldata eingebettet

optional: Childtag des eingebetteten XML-String

optional: Content für den Tag \$tagname

Output:

\$tagname

\$contentnew

\$parenttagname

XML-String Rückgabe des manipulierten XML-Strings

werden soll

3.9 Bibliothek Meta Data Generator

Diese Funktionsbibliothek ermöglicht Ihnen Keyword-Listen, die Description aus einem Inhalt zu erzeugen. Dies kann zur automatischen Erzeugung bzw. Befüllung von Metadaten verwendet werden.

Es können Meta Daten aus Multimedia-Dateien ausgelesen und im Container eines Objektes gespeichert werden.

3.9.1 getkeywords

Syntax:

getkeywords (\$text, \$language, \$charset)

Beschreibung:

Der Funktion wird der Inhalt übergeben. Damit werden alle Keywords aus dem Text ermittelt und als Keyword-Liste zurückgegeben.

Bsp

\$keywords = getkeywords ("This is just a short text.", "en", "UTF-8");

Input-Parameter:

\$text Content als String

\$language optional: Sprache [en, de], Standard ist "en" \$charset optional: Character Set, Standard ist "UTF-8"

Output:

Keywords Komma-getrennte Liste aller Keywords

False Fehler aufgetreten

3.9.2 getdescription

Syntax:

getdescription (\$text, \$charset)

Beschreibung:

Dieser Funktion wird der Inhalt übergeben. Daraufhin wird eine Kurzbeschreibung aus dem Text ermittelt und zurückgegeben.

Bsp:

\$keywords = getdescription ("This is just a short text.", "UTF-8");

Input-Parameter:

\$text Content als String

\$charset optional: Character Set, Standard ist "UTF-8"

Output:

Keywords Kurzbeschreibung des Inhaltes

3.9.3 injectmetadata

Syntax:

injectmetadata (\$site, \$location, \$object, \$mediafile, \$mapping, \$user)

Beschreibung:

Diese Funktion benötigt den Pfad zu einem Multimedia Objekt, den Dateiname des Objektes oder den Dateinamen der Multimedia-Datei sowie ein Mapping für die Speicherung der Daten. Damit wird der Text aus den Meta Daten anhand des Mappings ausgelesen und in der entsprechenden Text-ID des Containers geschrieben.

ACHTUNG: Bestehende Daten des Containers werden damit überschrieben!

```
Bsp:
```

```
// Mapping Definition (Meta Data Name -> Text-ID)
// Dublin Core
$mapping['dc: title'] = "Title";
$mapping['dc: subject'] = "Keywords";
$mapping['dc: description'] = "Description";
$mapping['dc: creator'] = "Creator";
$mapping['dc: rights'] = "Copyright";
// Adobe PhotoShop
$mapping['photoshop: SupplementalCategories'] = "Categories";
// Image Resolution defines Quality [Print, Web]
$mapping['hcms: quality'] = "Quality";
$result = injectmetadata ("Publication", "%comp%/test/", "image.jpg", "", $mapping, "Miller");
```

Input-Parameter:

\$site Name der Publikation

\$location Pfad im Filesystem (Position des Objektes)
\$object optional: Name des Objektes (Komponente)
\$mediafile oder optional: Name der Multimedia Datei
\$mapping Array (Meta Data Name -> Text-ID)

\$user Benutzername

Output:

True Meta Daten wurden erfolgreich gespeichert

3.10 Bibliothek Notifications

Diese Funktionsbibliothek versendet automatisierte Nachrichten an einen Benutzer anhand von Grenzwerten eines bestimmten Feldes.

Der Benutzer erhält eine vorformatierte Nachricht mit Information (Links) zu allen Objekten, die in den Suchbereich (Datumsober und -untergrenze) fallen.

3.10.1 licensenotification

Syntax:

licensenotification (\$site, \$cat, \$folderpath, \$text_id, \$date_begin, \$date_end, \$user)

Beschreibung:

Der Funktion ermittelt alle Objekte aufgrund des vorgegebenen Suchbereiches (Lokation und Datumsgrenzwerte) und versendet eine E-Mail an einen bestimmten Benutzer mit den Links zu allen betroffenen Objekten.

Bsp:

// set language for mail message
\$lang = "en";

// send mail to Miller

\$result = licensenotification ("Demo-DAM", "%comp%/images/", "comp", "valid_date",
"2012-09-01", "2012-09-30", "Miller");

Input-Parameter:

\$site Name der Publikation

\$cat Objekt Kategorie [page, comp]

\$folderpath Pfad für die Defintion des Suchbereiches

\$text_id Text ID des Feldes auf das die Suche angewendet werden soll

\$date_begin Startdatum für die Suche (YYYY-MM-DD) \$date_end Endedatum für die Suche (YYYY-MM-DD)

\$user Benutzername

Output:

True Mail wurde erfolgreich gesendet

4 Komponenten und Applikationen

Wenn Anwendungen in Komponenten integriert werden und Variablen aus einer Seite an eine Komponente übergeben werden müssen, so ist auf folgendes zu achten: Die Komponenten müssen über das Dateisystem eingebunden werden (nicht via HTTP). Alle Variablen die an die Komponente zu übergeben sind, sind in der Komponente als global zu definieren.

Bsp:

Eine Seite übergibt eine Variable an eine Komponente.

Hier der Code der Seite:

```
<html>
<head>
<title>page</title>
<head>
<body>
<?php $test="This is just a test!"; ?>
[hyperCMS:components id='component']
</body>
</html>
```

Der Code der Komponente muss wie folgt aussehen:

```
<?php
global $test;
echo $test;
?>
```

Im Beispiel wird die Variable \$test bzw. dessen Wert "This is just a test!" von der Komponente übernommen und in der Präsentation angezeigt.

5 Database Connectivity

Die Database Connectivity des hyper Content & Digital Asset Management Servers erlaubt die Anbindung von diversen Datenbanken zur Speicherung und Entnahme von Inhalten. Damit können z.B. relationale Datenbanken als externes Content Repository genutzt werden. Zu diesem Zweck ist je Template der entsprechende hyperCMS-Tag für die Database Connectivity einzufügen, der auf ein DB-Connect File verweißt.

In diesem File werden Funktionen hinterlegt, die hyperCMS aufruft, sofern das Template auf die Funktionsdatei zeigt.

Die Inhalte werden aus der Datenbank gelesen und dem Redakteur angezeigt. Verändert der Redakteur die Inhalte, so können diese auch wieder in die Datenbank geschrieben werden. Es können für Lese- und Schreibzugriffe auch verschiedene Datenbanken aufgerufen werden. Die Funktionen im DB-Connect File bieten nur die Hülle bzw. standardisierte Schnittstelle zu hyperCMS, die durch den Programmierer befüllt werden muss.

Das Thema der Datenbankintegration ist komplex und individuell zu behandeln, da auch bereits bestehende Datenbanken und deren Informationen integriert werden können. hyperCMS gibt kein ER-Modell vor bzw. legt sich auf keine speziellen Datenbank-Produkte fest. Generell kann gesagt werden, dass alle Möglichkeiten von PHP ausgeschöpft werden können, um sich zu diversen Datenquellen zu verbinden.

Neben den notwendigen Parameter für Queries auf relationale Datenbanken wird auch der gesamte Content Conatiner als XML-String übergeben. Damit könnten Dokumente bzw. Inhalte aus den Content Repository auch als Node in XML-Datenbanken abgelegt werden.

Sie selbst bestimmen, wohin Sie Ihre Daten speichern bzw. woher Sie diese holen. Mit PHP besitzen eine mächtige Sprache, die Ihnen Zugriff auf alle gängigen Datenbanken bietet.

Mehr Information zu den Funktionen von PHP finden Sie unter: http://www.php.net

5.1 Erstellen einer Database Connectivity

Möchten Sie eine Database Connectivity erstellen, so erstellen Sie eine Kopie des Files db_connect_default.php, dieses finden Sie in dem gewählten Root-Verzeichnis für die Ablage der Management Daten unter dem folgenden weiterführenden Pfad: /data/db_connect/ Die Kopie des Files nennen Sie z.B. nach der Datenbank, die Sie anbinden möchten.

Danach öffnen Sie die Datei und erhalten Einsicht in die Funktionen. Im Source Code finden Sie auch eine Beschreibung der Funktionen und der übergebenen Parameter als auch des Outputs.

Exemplarisch soll hier ein Lesezugriff auf eine MySQL Datenbank für einen Text-Inhalt dargestellt werden. Wir gehen davon aus, dass in einem Table "TextContent" die Inhalte mit dem Primary Key "container_id" und "text_id", dem Text-Inhalt "Text" sowie dem Text-Typ "Type" vorliegen. Der User sowie die Artikel ID wird nicht gesondert gespeichert, dies ist für die Eindeutigkeit des Inhalts auch nicht notwendig, denn die ID des Content Containers als auch die ID des Elements reichen als Primärschlüssel aus.

```
// ============== db connect ================================
// this file allows you to access a database using the full PHP functionality.
// you can read or write data from or into a database:
// the following parameter values are passed to each function for
// retrieving data from the database:
// name of the site: $site [string]
// name of the content container: $container_id [string] (is unique
// inside hyperCMS over all sites)
// content container: $container_content [XML-string]
// identification name: $id [string]
// ------ text ------
// if content is text
function db_read_text ($site, $content_id, $container_content, $id, $art_id, $user)
  // input variables: $id [string], optional: $artid [string], $user [string]
   // return value: $text [array]
                   the array must exactly look like this:
  //
                  $text[text], optional: $text[type]
constraints/accepted values for article type, see note below
  //
   // note: special characters in $text are escaped into
   // their html/xml equivalents.
               you can decide between unformatted, formatted and
   //
             optional text using $type:
   //
           unformatted text: $text[type] = textu
   //
   //
               formatted text: $text[type] = textf
   //
               text option: $text[type] = textl
   //-----
   $user = "username";
   $password = "password";
   $database = "database";
   // connect to database
   mysql_connect ("localhost", $user, $password);
   @mysql_select_db ($database) or die ("Unable to select database");
   // fire SQL-query
   $result = mysql_query ("SELECT Text, Type FROM TextContent WHERE
                 container_id=$container_id AND text_id=$id);
   // count returned rows, must be 1 if unique
   $num_of_rows = mysql_num_rows ($result);
   // get the result into an array namend $row
   if (\sum_{i=1}^{n} 
       $row = mysql_fetch_row ($result);
       // set values
       \text{stext[text]} = \text{srow[0]};
       \text{text[type]} = \text{row[1]};
   else $text = false;
   // close connection
   mysql_close ();
   // return result
  return $text;
```

6 Event System

Der hyper Content & Digital Asset Management Server beinhaltet ein Event System, das eine automatisierte Ausführung von Aktionen passierend auf Ereignissen im System ermöglicht. Damit lassen sich z.B. manuelle Vorgänge automatisieren.

Events werden meist durch den Benutzer durch Wahl einer Aktion gestartet, z.B. das Publizieren einer Seite. Ist der entsprechende Event aktiviert, so wird nach erfolgreicher Ausführung des Publikationsprozesses der Seite das Event "onpublishobject" aufgerufen. Die darin definierten Funktionen werden sodann ausgeführt.

Die Events des Event Systems können in der Datei "hypercms_eventsys.inc.php" definiert werden. Diese befindet sich im internen Repository im Ordner "eventsystem". In dieser Datei befinden sich auch weitere wichtige Hinweise, die bei der Ausführung von Events zu beachten sind.

Das Event System ist innerhalb des gesamten Management Systems über alle Publikationen gültig. Das System ist Bestandteil des hyperCMS APIs und wird somit bei jedem Aufruf einer Funktion des APIs ausgeführt.

Events lassen sich in der Datei "hypercms_eventsys.inc.php" aktivieren als auch deaktivieren, sodass der Einsatz der darin definierten Events leicht gesteuert werden kann.

Bei allen Events wird zwischen PRE- und POST-Events unterschieden. Das PRE-Event wird vor der eigentlichen Ausführung der aufgerufenen Aktion gestartet, während das POST-Event nach der erfolgreichen Ausführung der Aktion aufgerufen wird.

Bsp:

Beim Publizieren eines Objektes soll automatisch auch die Seite "index.php" die sich an der gleichen Position befindet publiziert werden, da diese z.B. ein über ein hyperCMS Script generiertes Verzeichnis aller Objekte des gleichen Ordners beinhaltet.

7 Liste der hyperCMS API Funktionen

Die Dokumentation aller API Funktionen sind auch auf unserer Website hypercms.com in der aktuellen Version verfügbar. Sie können die Dokumentation ihrer installierten Version als Hilfe und im Browser anzeigen. Nutzen Sie hierzu das ?-Icon im Template Editor um die Referenz aller hyperCMS Tags und API Funcktionen einzusehen.

7.1 Main API Functions

7.1.1 correctnumber

Syntax

correctnumber (\$number)

Input parameters

\$number ... formated number

global input parameters

\$mgmt_config

Output

correct mathematical number / false on error

7.1.2 cleancontent

Syntax

cleancontent (\$text, \$charset="UTF-8")

Input parameters

\$text ... text as string or array \$charset ... character set (optional)

global input parameters

\$mgmt_config

Output

cleaned text / false on error

Description

Removes all HTML tags, scripts and other special characters from the content in order to creaet a plain text

7.1.3 remove_utf8_bom

Syntax

remove_utf8_bom (\$text)

Input parameters

\$text ... text as string

Output

cleaned text / false on error

Description

Remove UTF-8 BOM sequences

7.1.4 convertchars

Syntax

convertchars (\$expression, \$charset_from="UTF-8", \$charset_to="UTF-8")

Input parameters

\$expression ... expression as string or array
\$charset_from ... input character set (optional)
\$charset_to ... output character set (optional)

global input parameters

\$mgmt_config

Output

converted expression / original expression on error

7.1.5 specialchr

Syntax

specialchr (\$expression, \$accept="")

Input parameters

\$expression ... expression \$accept ... list of characters to be excluded from search (optional)

Output

true/false

Description

Tests if an expression includes special characters (true) or does not (false). Allows you to accept characters through including it into \$accept (e.g. #\$...)

7.1.6 specialchr_encode

Syntax

specialchr_encode (\$expression, \$remove="no")

Input parameters

\$expression ... expression \$remove ... remove all special characters [yes, no]

global input parameters

\$user
\$mgmt_config
\$hcms_lang
\$lang

Output

expression without special characters (for file names)

Description

Renames all special characters for file names to an expression according to given rules

7.1.7 specialchr_decode

Syntax

specialchr_decode (\$expression)

Input parameters

\$expression ... expression

global input parameters

\$user
\$mgmt_config
\$hcms_lang
\$lang

Output

expression with special characters (for file names) / false

Description

This is the decode function for function specialchr_encode

7.1.8 object_exists

Syntax

object_exists (\$path)

Input parameters

\$path ... path to an object

global input parameters

\$user
\$mgmt_config
\$hcms_lang
\$lang

Output

true / false

Description

This function verifies if an object exists already.

7.1.9 is_utf8

Syntax

is_utf8 (\$str)

Input parameters

\$str ... expression

Output

if string is utf-8 encoded true / false otherwise

Description

This function is an alternative to mb_check_encoding (which requires an extra PHP module). It is not failsave!

7.1.10 is_latin1

Syntax

is_latin1 (\$str)

Input parameters

\$str ... expression

Output

if string is latin 1 encoded true / false otherwise

Description

This function is an alternative to mb_check_encoding (which requires an extra PHP module). It is not failsave!

7.1.11 makestring

Syntax

makestring (\$array)

Input parameters

\$array ... (multidimensional) array

Output

string including all array values / false on error

7.1.12 splitstring

Syntax

splitstring (\$string)

Input parameters

\$string ... string with ";" or ", " as seperator

Output

array with string splitted into array / false on error

7.1.13 is_emptyfolder

Syntax

is_emptyfolder (\$dir)

Input parameters

\$dir ... path to folder

global input parameters

\$mgmt_config

Output

true / false

Description

Checks if a directory/folder is empty (has no published objects or other files)

7.1.14 is_supported

Syntax

is_supported (\$preview_array, \$file)

Input parameters

\$preview_array ... preview array holding the supported file extensions as key and references to executables as value

\$file ... file name or file extension

Output

true / false

Description

This function determines if a certain file type by its file extension is supported by the systems media conversion

7.1.15 is_cloudstorage

Syntax

is_cloudstorage (\$site="")

Input parameters

\$site ... publication name (optional)

global input parameters

\$mgmt_config

Output

true / false

Description

This function determines if a cloud storage has been defined in the main configuration or for a specific publication

7.1.16 is_cloudobject

Syntax

is_cloudobject (\$file)

Input parameters

\$file ... path to media file or media file name

global input parameters

\$mgmt_config

Output

true / false

Description

This functions verifies if an object/file is available in the cloud storage

7.1.17 is_date

Syntax

is_date (\$date, \$format="Y-m-d")

Input parameters

\$date ... date as string \$format ... date format (optional)

Output

true / false

Description

This function determines if a string represents a valid date format

7.1.18 is_tempfile

Syntax

is_tempfile (\$path)

Input parameters

\$path ... file name or path [string]

global input parameters

\$user
\$mgmt_config
\$hcms_lang
\$lang

Output

if file is a temp file true / false on error

Description

This functions checks if the provided file name is a temporary file

7.1.19 is_thumbnail

Syntax

is_thumbnail (\$media, \$images_only=true)

Input parameters

\$media ... file name or path [string]

\$images_only ... only thumbnail images should be considered as thumbnail [true, false]

global input parameters

\$user
\$mgmt_config
\$hcms_lang
\$lang

Output

if file is a thumbnail file true / false on error

Description

This functions checks if the provided file name is a thumbnail file

7.1.20 is_config

Syntax

is_config (\$media)

Input parameters

\$media ... file name or path [string]

global input parameters

\$user \$mgmt_config \$hcms_lang \$lang

Output

if file is a config file true / false if not

Description

This functions checks if the provided file name is a config file

7.1.21 is_encryptedfile

Syntax

is_encryptedfile (\$location, \$file)

Input parameters

\$location ... path to file [string] \$file ... file name [string]

global input parameters

\$user
\$mgmt_config
\$hcms_lang
\$lang

Output

true / false

Description

This functions checks if the provided file is encrypted

7.1.22 is_document

Syntax

is_document (\$file)

Input parameters

\$file ... file name or file extension

global input parameters

\$mgmt_config \$hcms_ext

Output

true / false

Description

This function determines if a certain file is a document (binary and text based)

7.1.23 is_image

Syntax

is_image (\$file)

Input parameters

\$file ... file name or file extension

global input parameters

\$mgmt_config \$hcms_ext

Output

true / false

Description

This function determines if a certain file is an image

7.1.24 is_rawimage

Syntax

is_rawimage (\$file)

Input parameters

\$file ... file name or file extension

global input parameters

\$mgmt_config \$hcms_ext

Output

true / false

Description

This function determines if a certain file is a raw image

7.1.25 is_aiimage

Syntax

is_aiimage (\$file)

Input parameters

\$file ... file name or file extension

Output

true / false

Description

This function determines if a certain file is a vector-based Adobe Illustrator (AI) or AI-compatible EPS file

7.1.26 is_video

Syntax

is_video (\$file)

Input parameters

\$file ... file name or file extension

global input parameters

\$mgmt_config \$hcms_ext

Output

true / false

Description

This function determines if a certain file is a video

7.1.27 is_audio

Syntax

is_audio (\$file)

Input parameters

\$file ... file name or file extension

global input parameters

\$mgmt_config
\$hcms_ext

Output

true / false

Description

This function determines if a certain file is an audio file

7.1.28 is_mobilebrowser

Syntax

is_mobilebrowser ()

Input parameters

global input parameters

\$user

\$mgmt_config

Output

true / false

Description

Detects mobile browsers (smartphones and tablets)

7.1.29 is_iOS

Syntax

is_iOS ()

Input parameters

global input parameters

\$user

\$mgmt_config

Output

true / false

Description

Detects if a mobile browser is an iPhone, iPad or IPod

7.1.30 copyrecursive

Syntax

copyrecursive (\$src, \$dst)

Input parameters

\$src ... source directory

\$dst ... destination directory

Output

true/false

Description

This function copyies all directories and files from source to destination directory

7.1.31 array_iunique

Syntax

array_iunique (\$array)

Input parameters

\$array ... array

Output

unique array / false

Description

This function is the case-insensitive form of PHPs array_unique function

7.1.32 in_array_substr

Syntax

in_array_substr (\$search, \$array)

Input parameters

\$search ... search expression \$array ... array

Output

true / false

Description

This function is supporting the search of substrings in the array values compared to PHPs in_array function

7.1.33 createfilename

Syntax

createfilename (\$filename)

Input parameters

\$filename ... path to file or directory

global input parameters

\$mgmt_config \$hcms_lang \$lang

Output

new filename/false

Description

Create an valid file name without special characters that does not exceed the maximum file name length

7.1.34 correctfile

Syntax

correctfile (\$abs_path, \$filename, \$user="")

Input parameters

\$abs_path ... path to file or directory \$filename ... file or directory name \$user ... user name

global input parameters

\$mgmt_config
\$hcms_lang
\$lang

Output

correct filename/false

7.1.35 correctpath

Syntax

correctpath (\$path, \$slash="/")

Input parameters

\$path ... path to folder
\$slash ... directory seperator (optional)

Output

correct path/false

7.1.36 convertpath

Syntax

convertpath (\$site, \$path, \$cat="")

Input parameters

\$site ... publication\$path ... content management path to folder or object\$cat ... object category ['page

global input parameters

\$user
\$mgmt_config
\$hcms_lang
\$lang

Output

converted path or URL / false on error

Description

This function replaces object pathes of the content management config with %page% and %comp% path variables

7.1.37 convertlink

Syntax

convertlink (\$site, \$path, \$cat)

Input parameters

\$site ... publication

\$path ... publication management path to folder or object

\$cat ... object category ['page

global input parameters

\$user \$mgmt_config \$publ_config \$hcms_lang \$lang

Output

converted path or URL / false on error

Description

This function replaces pathes of the publication management config with %page% and %comp% path variables.

7.1.38 deconvertpath

Syntax

deconvertpath (\$objectpath, \$type="file", \$specialchr_transform=true)

Input parameters

\$objectpath ... string including path to folder or object \$type ... convert to file system path or URL [file \$specialchr_transform ... url] (optional)

global input parameters

\$user
\$mgmt_config
\$hcms_lang
\$lang

Output

deconverted path/false

Description

This function replaces all %page% and %comp% path variables with the path of the content management config.

It converts the path only on content management side not for the publication target. It optionally transform special characters as well.

BE AWARE: The input path must not provide template data since valid_publicationname mightreturn false.

7.1.39 deconvertlink

Syntax

deconvertlink (\$path, \$type="url")

Input parameters

\$path ... path to folder or object

\$type ... convert to file system path or URL [file

global input parameters

\$user

\$mgmt_config
\$publ_config
\$hcms_lang
\$lang

Output

converted absolute link without host/false

Description

This function deconverts the path only for the publication target.

It should be used for page linking, otherwise the function will return the absolute link including the host for component links.

7.1.40 createviewlink

Syntax

createviewlink (\$site, \$mediafile, \$name="", \$force_reload=false, \$type="wrapper")

Input parameters

```
$site ... publication
$mediafile ... media file name
$name ... media name (optional)
$force_reload ... force reload [true, false] (optional)
$type ... link type [wrapper, download] (optional)
```

global input parameters

\$user

\$mgmt_config

Output

URL for download of the multimedia file / false on error

Description

The view link is mainly used inside the system in order to reference and load a multimedia file.

The database is not required since the object hash or ID is not needed to create the view link.

7.1.41 createaccesslink

Syntax

```
createaccesslink ($site, $location="", $object="", $cat="", $object_id="", $login, $type="al",
$lifetime=0, $formats="")
```

Input parameters

```
$site ... publication

$location ... location (optional)

$object ... object (optional)

$cat ... category [page, comp] (optional)

$object_id ... object-ID (optional)

$login ... user login

$type ... link type [al, dl] (optional)

$lifetime ... token lifetime in seconds (optional)

$formats ... formats (optional)
```

global input parameters

\$user \$mgmt_config \$hcms_lang

\$lang

Output

URL for access to given object / false on error

7.1.42 createobjectaccesslink

Syntax

```
createobjectaccesslink ($site="", $location="", $object="", $cat="", $object_id="",
$container_id="")
```

Input parameters

```
$site ... publication (optional)
$location ... location (optional)
$object ... object (optional)
$cat ... category [page, comp] (optional)
$object_id ... object ID (optional)
$container_id ... container-ID (optional)
```

global input parameters

\$user
\$mgmt_config
\$hcms_lang
\$lang

Output

URL for download of the multimedia file of the given object or folder / false on error

7.1.43 createwrapperlink

Syntax

```
createwrapperlink ($site="", $location="", $object="", $cat="", $object_id="", $container_id="")
```

Input parameters

```
$site ... publication (optional)
$location ... location (optional)
$object ... object (optional)
$cat ... category [page, comp] (optional)
$object_id ... object ID (optional)
$container_id ... container-ID (optional)
```

global input parameters

```
$user
$mgmt_config
$hcms_lang
$lang
```

Output

URL for download of the multimedia file of the given object or folder / false on error

Description

In order to track and include external user IDs in the daily statistics you need to manually add the 'user' parameter to the link in the form of: &user=[user-ID]

7.1.44 createdownloadlink

Syntax

```
createdownloadlink ($site="", $location="", $object="", $cat="", $object_id="",
$container_id="")
```

Input parameters

\$site ... publication name (optional)
\$location ... location (optional)
\$object ... object (optional)
\$cat ... category [page, comp] (optional)
\$object_id ... object ID (optional)
\$container_id ... container-ID (optional)

global input parameters

\$user
\$mgmt_config
\$hcms_lang
\$lang

Output

URL for download of the multimedia file of the given object or folder / false on error

Description

In order to track and include external user IDs in the daily statistics you need to manually add the 'user' parameter to the link in the form of: &user=[user-ID]

7.1.45 createmultidownloadlink

Syntax

createmultidownloadlink (\$site, \$multiobject="", \$media="", \$location="", \$name="", \$user, \$type="", \$mediacfg="", \$linktype="download")

Input parameters

\$site ... publication name
\$multiobject ... multiobject string (optional)
\$media ... media file name (optional)
\$location ... location (optional)
\$name ... presentation name (optional)
\$user ... user name
\$type ... conversion type (format
\$mediacfg ... e.g. jpg)
\$linktype ... media configuration used for conversion (e.g.: 1024x768px)

global input parameters

\$mgmt_config \$mgmt_compress \$pageaccess \$compaccess \$hiddenfolder \$hcms_linking \$globalpermission \$setlocalpermission \$hcms_lang \$lang

Output

download link / false on error

Description

Generates a download link of a single media file, folder or multi objects.

Priority if multiple input parameters for media file, folder or multi objects are given:

1st...multi objects

2nd...media file

3rd...folder

7.1.46 cleandomain

Syntax

cleandomain (\$path)

Input parameters

\$path ... string to clean from http(s)://domain

global input parameters

\$mgmt_config \$hcms_lang \$lang

Output

cleanded string / false on error

Description

Returns the URL notation without the protocoll://domain.

7.1.47 fileversion

Syntax

fileversion (\$file)

Input parameters

\$file ... file name [string]

Output

versioned file name [string] / false on error

Description

Creates a version file name

7.1.48 createversion

Syntax

createversion (\$site, \$file, \$user="sys")

Input parameters

\$site ... publication name

\$file ... media file name or container name

\$user ... user name (optional)

global input parameters

\$mgmt_config
\$mgmt_mediaoptions
\$mgmt_docoptions
\$hcms_ext
\$user

Output

true / false

Description

Creates a new version of a multimedia file and container

7.1.49 rollbackversion

Syntax

rollbackversion (\$site, \$location, \$page, \$container_version, \$user="sys")

Input parameters

\$site ... publication name
\$location ... location
\$page ... object name
\$container_version ... container version name
\$user ... user name (optional)

global input parameters

\$mgmt_config
\$mgmt_mediaoptions
\$mgmt_docoptions
\$hcms_ext
\$hcms_lang
\$lang

Output

result_array

Description

Makes an older object version to the current version

7.1.50 deleteversion

Syntax

deleteversion (\$site, \$container_version, \$user="sys")

Input parameters

\$site ... publication name \$container_version ... location \$user ... object name

global input parameters

\$mgmt_config

Output

true / false

Description

Removes the version of an object

7.1.51 deleteversions

Syntax

deleteversions (\$type, \$report, \$user="sys")

Input parameters

\$type ... type [content, template] or valid path in filesystem
\$report ... report [yes, no]
\$user ... user name (optional)

global input parameters

\$user
\$mgmt_config
\$hcms_lang
\$lang

Output

true [report=no] or report [report=yes]
false on error

Description

Removes all versions of all objects or templates

7.1.52 loadfile_header

Syntax

loadfile_header (\$abs_path, \$filename)

Input parameters

\$abs_path ... path to file \$filename ... file name

global input parameters

\$user \$mgmt_config \$hcms_lang \$lang

Output

file content

Description

Loads the file header, represented by a defined header size.

7.1.53 loadfile_fast

Syntax

loadfile_fast (\$abs_path, \$filename)

Input parameters

\$abs_path ... path to file \$filename ... file name

global input parameters

\$user
\$mgmt_config
\$hcms_lang
\$lang

Output

file content

Description

This functions is identical to loadfile, but it does not wait for locked files to be unlocked again. It should only be used on files that won't be locked by the system. It is therefore recommended to use loadfile.

7.1.54 loadfile

Syntax

loadfile (\$abs_path, \$filename)

Input parameters

\$abs_path ... path to file \$filename ... file name

global input parameters

\$user
\$mgmt_config
\$hcms_lang
\$lang

Output

file content

Description

This function loads a file and waits up to 3 seconds for locked files to be unlocked

7.1.55 loadlockfile

Syntax

loadlockfile (\$user, \$abs_path, \$filename, \$force_unlock=3)

Input parameters

\$user ... user
\$abs_path ... path to file
\$filename ... file name
\$force_unlock ... force unlock of file after x seconds [integer]

global input parameters

\$mgmt_config
\$hcms_lang
\$lang

Output

file content

Description

This function loads and locks a file for a sepecific user. It waits up to 3 seconds for locked files to be unlocked.

Function loadlockfile and savelockfile includes a locking mechanismen for files.

Every time you want to lock a file during your operations use loadlockfile.

It is important to use savelockfile to save and unlock the file again.

savelockfile requires the file to be opened by loadlockfile before.

7.1.56 savefile

Syntax

savefile (\$abs_path, \$filename, \$filedata)

Input parameters

\$abs_path ... path to file \$filename ... file name \$filedata ... file content

global input parameters

\$user
\$mgmt_config
\$hcms_lang
\$lang

Output

true/false

Description

This function saves content to a file

7.1.57 savelockfile

Syntax

savelockfile (\$user, \$abs_path, \$filename, \$filedata)

Input parameters

\$user ... user \$abs_path ... path to file \$filename ... file name \$filedata ... file content

global input parameters

\$mgmt_config \$hcms_lang \$lang

Output

true/false

Description

Saves content to a locked file. It requires the file to be opened by loadlockfile. Function loadlockfile and savelockfile includes a locking mechanismen for files. Every time you want to lock a file during your operations use loadlockfile. It is important to use savelockfile to save and unlock the file again. savelockfile requires the file to be opened by loadlockfile before.

7.1.58 appendfile

Syntax

appendfile (\$abs_path, \$filename, \$filedata)

Input parameters

\$abs_path ... path to file \$filename ... file name \$filedata ... file content

global input parameters

\$user
\$mgmt_config
\$hcms_lang
\$lang

Output

true/false

Description

Appends data to a file but cannot create a new file! Waits up to 3 seconds for locked files to be unlocked again. Files won't be unlocked if the file is already locked.

7.1.59 lockfile

Syntax

lockfile (\$user, \$abs_path, \$filename)

Input parameters

\$user ... user \$abs_path ... path to file \$filename ... file name

global input parameters

\$mgmt_config
\$hcms_lang
\$lang

Output

true/false

Description

This functions lockes a file for a specific user

7.1.60 unlockfile

Syntax

unlockfile (\$user, \$abs_path, \$filename)

Input parameters

\$user ... user \$abs_path ... path to file \$filename ... file name

global input parameters

\$mgmt_config
\$hcms_lang
\$lang

Output

true/false

Description

This functions unlockes a file for a specific user

7.1.61 deletefile

Syntax

deletefile (\$abs_path, \$filename, \$recursive=false)

Input parameters

\$abs path ... path to file

\$filename ... file or directory name

\$recursive ... delete all files in directory recursively including symbolic links [true, false]

global input parameters

\$user

\$mgmt_config
\$hcms_lang
\$lang

Output

true/false

Description

Deletes a file or directory. If parameter recursive is et to true all items of a directory will be removed as well.

7.1.62 restoremediafile

Syntax

restoremediafile (\$site, \$mediafile)

Input parameters

\$site ... publication name \$mediafile ... media file name

global input parameters

\$mgmt_config

Output

result array

Description

Moves an exported media file back to the media repository

7.1.63 preparemediafile

Syntax

preparemediafile (\$site, \$medialocation, \$mediafile, \$user="")

Input parameters

\$site ... publication name \$medialocation ... media file location \$mediafile ... media file name \$user ... user name (optional)

global input parameters

\$mgmt_config

Output

result array / false on error

Description

Prepares a media file for use in the system (load from cloud, decrypt content)

7.1.64 deletemediafiles

Syntax

deletemediafiles (\$site, \$mediafile, \$delete_original=false)

Input parameters

\$site ... publication name\$mediafile ... mediafile name\$delete_original ... delete original media file [true, false] (optional)

global input parameters

\$user
\$mgmt_config
\$mgmt_mediaoptions
\$mgmt_docoptions
\$hcms_ext
\$hcms_lang
\$lang

Output

true/false

Description

Deletes all derivates (thumbnails, config files, converted versions of the file) of a specific media file resource. Deletes the original media file optionally .

7.1.65 avoidfilecollision

Syntax

avoidfilecollision (\$data="tempdata", \$force=false)

Input parameters

\$data ... data string (optional) \$force ... force execution [true, false]

global input parameters

\$mgmt_config \$site

Output

true / false on error

Description

Appending data to a file ensures that the previous write process is finished (required due to issue when editing encrypted files)

7.1.66 substr_in_array

Syntax

substr_in_array (\$search, \$array)

Input parameters

\$search ... search-string \$array ... array

Output

array with found values / false

Description

Searches for substring in array

7.1.67 downloadobject

Syntax

downloadobject (\$location, \$object, \$container="", \$lang="en", \$user="")

Input parameters

\$location ... location \$object ... object name \$container ... content container \$lang ... language (optional) \$user ... user name (optional)

global input parameters

\$mgmt_config \$eventsystem \$hcms_lang \$lang

Output

stream of file content / false on error

Description

This functions provides an object via http for viewing, not suitable for multimedia objects!

7.1.68 downloadfile

Syntax

downloadfile (\$filepath, \$name, \$force="wrapper", \$user="")

Input parameters

\$filepath ... path to file [string]

\$name ... file name to show for download via http

\$force ... force file wrapper or download or no file headers for WebDAV [download, wrapper, noheader]

\$user ... user name (optional)

global input parameters

\$mgmt_config \$eventsystem \$hcms_lang \$lang \$is_iphone

Output

stream of file content / false on error

Description

This functions provides a file via http for view or download

7.1.69 loadcontainer

Syntax

loadcontainer (\$container, \$type="work", \$user)

Input parameters

\$container ... container file name or container id (working container will be loaded by default)
\$type ... optional container type [published
\$user ... work

global input parameters

\$mgmt_config
\$hcms_lang
\$lang

Output

XML content of container / false on error

Description

This functions loads a content container

7.1.70 savecontainer

Syntax

savecontainer (\$container, \$type="work", \$data, \$user, \$init=false)

Input parameters

\$container ... container file name or container id (working container will be loaded by default)

\$type ... optional container type [published, work, version]

\$data ... container content

\$user ... user

\$init ... save container initally [true, false] (optional)

global input parameters

\$mgmt_config
\$hcms_lang
\$lang

Output

true / false on error

Description

Saves data into existing content container by default. Only if \$init is set to true it will initally save a non existing container.

7.1.71 checkworkflow

Syntax

checkworkflow (\$site, \$location, \$page, \$cat="", \$contentfile="", \$contentdata="", \$buildview="cmsview", \$viewstore="", \$user)

Input parameters

\$site ... publication name [string]
\$location ... location [string]
\$page ... object name [string]
\$cat ... category [page, comp] (optional)
\$contentfile ... container name [string] (optional)
\$contentdata ... container [XML string] (optional)
\$buildview ... view name [string] (optional)
\$viewstore ... view store [string] (optional)
\$user ... user name [string]

global input parameters

\$mgmt_config

Output

result array

Description

Help function for function buildview to evaluate the workflow of an object and return the manipulated view store, view name, workflow ID, workflow role and the encrypted workflow token.

7.1.72 inherit_db_load

Syntax

inherit_db_load (\$user)

Input parameters

\$user ... nothing

global input parameters

\$siteaccess \$mgmt_config \$hcms_lang \$lang

Output

inheritance database [2 dim. array]/false

Description

This function loads and locks the inheritance database.

Each record of the inherit management database has the following design:

xml-content container : | absolute path to 1-n objects : | 1-m inherits used by 1-n objects Important: The inherit management database needs to be saved or closed after loading it

7.1.73 inherit_db_read

Syntax

inherit_db_read ()

Input parameters

global input parameters

\$user
\$mgmt_config
\$hcms_lang
\$lang

Output

inheritance database [2 dim. array]/false

Description

This function loads the inheritance database for reading

7.1.74 inherit_db_close

Syntax

inherit_db_close (\$user)

Input parameters

\$user ... %

global input parameters

\$mgmt_config \$hcms_lang \$lang

Output

true/false

Description

Closes and unlocks the inheritance management database

7.1.75 inherit_db_save

Syntax

inherit_db_save (\$inherit_db, \$user)

Input parameters

\$inherit_db ... inherit database array \$user

global input parameters

\$mgmt_config
\$hcms_lang
\$lang

Output

true/false

Description

This function saves und unlocks the inheritance management database

7.1.76 inherit_db_getparent

Syntax

inherit_db_getparent (\$inherit_db, \$child)

Input parameters

\$inherit_db ... inherit database [2 dim. array] \$child ... child [string]

Output

all parents of given child [1 dim. array] / false

7.1.77 inherit_db_getchild

Syntax

inherit_db_getchild (\$inherit_db, \$parent)

Input parameters

\$inherit_db ... parent [string] \$parent

Output

all childs of given parent [1 dim. array] / false

7.1.78 inherit_db_setparent

Syntax

inherit_db_setparent (\$inherit_db, \$child, \$parent_array)

Input parameters

\$inherit_db ... inherit database [2 dim. array] \$child ... child [string] \$parent_array ... parents [Array]

Output

inherit database [2 dim. array]

Description

This function updates and insert all references from a child an its parents

7.1.79 inherit_db_insertparent

Syntax

inherit_db_insertparent (\$inherit_db, \$parent, \$child_array)

Input parameters

\$inherit_db ... inherit database [2 dim. array]
\$parent ... parent [string]
\$child_array ... childs [Array]

Output

inherit database [2 dim. array]

7.1.80 inherit_db_deleteparent

Syntax

inherit_db_deleteparent (\$inherit_db, \$parent)

Input parameters

\$inherit_db ... inherit database [2 dim. array] \$parent ... parent [string]

Output

inherit database [2 dim. array]

7.1.81 createinstance

Syntax

createinstance (\$instance_name, \$settings, \$user="sys")

Input parameters

\$instance_name ... instance name \$settings ... settings array \$user ... user name

global input parameters

\$eventsystem
\$mgmt_config
\$hcms_lang
\$lang

Output

result array

Description

This function creates a new instance with all its files and the mySQL database

7.1.82 editinstance

Syntax

editinstance (\$instance_name, \$content, \$user="sys")

Input parameters

\$instance_name ... instance name \$content ... content as string \$user ... user name

global input parameters

\$eventsystem
\$mgmt_config
\$hcms_lang
\$lang

Output

result array

Description

This function saves the instance configuration in the config file

7.1.83 deleteinstance

Syntax

deleteinstance (\$instance_name, \$user="sys")

Input parameters

\$instance_name ... instance name \$user ... settings array

global input parameters

\$eventsystem
\$mgmt_config
\$hcms_lang
\$lang

Output

result array

Description

This function creates a new instance with all its files and the mySQL database

7.1.84 createpublication

Syntax

createpublication (\$site_name, \$user="sys")

Input parameters

\$site_name ... publication name \$user ... user name

global input parameters

\$eventsystem
\$mgmt_config
\$hcms_lang
\$lang

Output

result array

Description

This function creates a new publication with all its files

7.1.85 editpublication

Syntax

editpublication (\$site_name, \$setting, \$user="sys")

Input parameters

\$site_name ... publication name \$setting ... publication settings array \$user ... user name

global input parameters

\$eventsystem \$mgmt_config \$hcms_lang \$lang

Output

result array

Description

This function saves the settings of a publication

7.1.86 editpublicationsetting

Syntax

editpublicationsetting (\$site_name, \$setting, \$value, \$user="sys")

Input parameters

\$site_name ... publication name
\$setting ... publication settings name (see publication config file for details)
\$value ... value
\$user ... user name

global input parameters

\$eventsystem
\$mgmt_config
\$hcms_lang
\$lang

Output

true/false

Description

This function can be used to edit a single settings of a publication

7.1.87 deletepublication

Syntax

deletepublication (\$site_name, \$user="sys")

Input parameters

\$site_name ... publication name
\$user ... user name

global input parameters

\$mgmt_config
\$eventsystem
\$hcms_lang
\$lang

Output

result array

Description

This function deletes a publication with all its files

7.1.88 createpersonalization

Syntax

createpersonalization (\$site, \$pers_name, \$cat)

Input parameters

\$site ... publication name
\$pers_name ... personalization profile or tracking name
\$cat ... category [profile, tracking]

global input parameters

\$user \$eventsystem \$mgmt_config \$hcms_lang \$lang

Output

result array

Description

This function creates a new customer personalization tracking or profile

7.1.89 deletepersonalization

Syntax

deletepersonalization (\$site, \$pers_name, \$cat)

Input parameters

\$site ... publication name
\$pers_name ... personalization profile or tracking name
\$cat ... category [profile, tracking]

global input parameters

\$user \$eventsystem \$mgmt_config \$hcms_lang \$lang

Output

result array

Description

This function deletes a customer personalization tracking or profile

7.1.90 createtemplate

Syntax

createtemplate (\$site, \$template, \$cat)

Input parameters

\$site ... publication name
\$template ... template name
\$cat ... category [page, comp, meta, inc]

global input parameters

\$user \$eventsystem \$mgmt_config \$hcms_lang \$lang

Output

result array

Description

This function creates a new template

7.1.91 loadtemplate

Syntax

loadtemplate (\$site, \$template)

Input parameters

\$site ... publication name \$template ... template file name

global input parameters

\$user
\$mgmt_config
\$hcms_lang
\$lang

Output

array (template content [XML string] publication result[true/false]) / false on error

Description

This function loads templates by given name.

Based on the inheritance settings of the publication the template will be loaded with highest priority from the own publication and if not available from a parent publication. If the parent publications have double entries the sort mechanism will define the priority. First priority have numbers, second are upper case letters and last priority have lower case letters.

7.1.92 edittemplate

Syntax

edittemplate (\$site, \$template, \$cat, \$user, \$content="", \$extension="", \$application="")

Input parameters

\$site ... publication name

\$template ... template file name \$cat ... category [page, comp, meta, inc] \$user ... template content (optional) \$content ... template extension (optional) \$extension ... temlate application (optional) \$application

global input parameters

\$eventsystem
\$mgmt_config
\$hcms_lang
\$lang

Output

result array

Description

This function edites a template

7.1.93 deletetemplate

Syntax

deletetemplate (\$site, \$template, \$cat)

Input parameters

\$site ... publication name\$template ... template file name\$cat ... category [page, comp, meta, inc]

global input parameters

\$user \$eventsystem \$mgmt_config \$hcms_lang \$lang

Output

result array

Description

This function deletes a template

7.1.94 createuser

Syntax

createuser (\$site, \$login, \$password, \$confirm_password, \$user="sys")

Input parameters

\$site ... publication name (optional)
\$login ... login name
\$password ... password
\$confirm_password ... confirmed password
\$user ... user name

global input parameters

\$eventsystem
\$mgmt_config
\$mgmt_lang_shortcut_default

```
$hcms_lang
$lang
```

Output

array

Description

This function creates a new user

7.1.95 edituser

Syntax

```
edituser ($site, $login, $old_password="", $password="", $confirm_password="", $superadmin="0", $realname="", $language="en", $theme="", $email="", $phone="", $signature="", $usergroup="", $usersite="", $user="sys")
```

Input parameters

```
$site ... publication name
$login ... login name
$old_password ... new login name
$password ... password
$confirm_password ... confirmed password
$superadmin ... super administrator [0, 1]
$realname ... real name
$language ... language setting [de, en]
$theme ... theme name (optional)
$email ... email
$phone ... phone
$signature ... usergroup string [group1|group2]
$usergroup ... member of site(s) string [site1|site2]]
$usersite ... user name
$user
```

global input parameters

\$eventsystem \$login_cat \$group \$mgmt_config \$hcms_lang \$lang

Output

array

Description

This function edits a user

7.1.96 deleteuser

Syntax

deleteuser (\$site, \$login, \$user="sys")

Input parameters

```
$site ... publication where the user should be removed [*Null*] for all publications $login ... login name $user ... user name
```

global input parameters

\$eventsystem
\$mgmt_config
\$hcms_lang
\$lang

Output

array

Description

This function removes a user

7.1.97 creategroup

Syntax

creategroup (\$site, \$group_name, \$user="sys")

Input parameters

\$site ... publication anem \$group_name ... group name \$user ... user name

global input parameters

\$eventsystem
\$mgmt_config
\$hcms_lang
\$lang

Output

array

Description

This function creates a new user group

7.1.98 editgroup

Syntax

editgroup (\$site, \$group_name, \$pageaccess, \$compaccess, \$permission, \$user)

Input parameters

\$site ... publication name
\$group_name ... group name
\$pageaccess ... page folder access array
\$compaccess ... component folder access array
\$permission ... permissions array
\$user ... user name

global input parameters

\$eventsystem
\$mgmt_config
\$hcms_lang
\$lang

Output

array

Description

This function removes a user group

7.1.99 deletegroup

Syntax

deletegroup (\$site, \$group_name, \$user)

Input parameters

\$site ... publication name \$group_name ... group name \$user ... user name

global input parameters

\$eventsystem
\$mgmt_config
\$hcms_lang
\$lang

Output

array

Description

This function removes a user group

7.1.100 renamegroupfolder

Syntax

renamegroupfolder (\$site, \$cat, \$folder_curr, \$folder_new, \$user)

Input parameters

\$site ... publication name \$cat ... cat[page, comp] \$folder_curr ... old location \$folder_new ... new location \$user ... user

global input parameters

\$mgmt_config \$hcms_lang \$lang

Output

true / false on error

Description

This function renames a workgroup folder

7.1.101 deletegroupfolder

Syntax

deletegroupfolder (\$site, \$cat, \$folderpath, \$user)

Input parameters

\$site ... publication name \$cat ... cat[page, comp] \$folderpath ... path to the folder \$user ... user

global input parameters

\$mgmt_config

\$hcms_lang \$lang

Output

true / false on error

Description

This function removes a group folder

7.1.102 renameworkflowfolder

Syntax

renameworkflowfolder (\$site, \$cat, \$folder_curr, \$folder_new, \$user)

Input parameters

\$site ... publication name \$cat ... cat[page, comp] \$folder_curr ... old location \$folder_new ... new location \$user ... user

global input parameters

\$mgmt_config \$hcms_lang \$lang

Output

true / false on error

Description

This function renames a workgroup folder

7.1.103 deleteworkflowfolder

Syntax

deleteworkflowfolder (\$site, \$cat, \$folderpath, \$user)

Input parameters

\$site ... publication name
\$cat ... cat[page, comp]
\$folderpath ... location of folder
\$user ... user

global input parameters

\$mgmt_config
\$hcms_lang
\$lang

Output

true / false on error

Description

This function removes a workgroup folder

7.1.104 createmediacat

Syntax

createmediacat (\$site, \$mediacat_name)

Input parameters

\$site ... publication

\$mediacat_name ... media category name

global input parameters

\$user \$eventsystem \$mgmt_config \$hcms_lang \$lang

Output

Array with onload JS-code and message

Description

Creates a new media category in the template media index file. Only used for template media.

7.1.105 renamemediacat

Syntax

renamemediacat (\$site, \$mediacat_name_curr, \$mediacat_name)

Input parameters

\$site ... publication \$mediacat_name_curr ... old media category name \$mediacat_name ... new media category name

global input parameters

\$user
\$eventsystem
\$mgmt_config
\$hcms_lang
\$lang

Output

Array with onload JS-code and message

Description

Renames a new media category in the template media index file

7.1.106 deletemediacat

Syntax

deletemediacat (\$site, \$mediacat_name)

Input parameters

\$site ... publication

\$mediacat_name ... media category name

global input parameters

\$user \$eventsystem \$mgmt_config \$hcms_lang \$lang

Output

Array with onload JS-code and message

Description

Deletes a new media category in the template media index file

7.1.107 uploadtomediacat

Syntax

uploadtomediacat (\$site, \$mediacat_name, \$global_files)

Input parameters

\$site ... publication \$mediacat_name ... media category name \$global_files ... PHP FILES array

global input parameters

\$user \$eventsystem \$mgmt_config \$hcms_lang \$lang

Output

Array with onload JS-code and message

Description

Uploads a media file into a given template media category

7.1.108 deletefrommediacat

Syntax

deletefrommediacat (\$site, \$mediafile)

Input parameters

\$site ... publication \$mediafile ... media file name

global input parameters

\$user \$eventsystem \$mgmt_config \$hcms_lang \$lang

Output

Array with onload JS-code and message

Description

Deletes a media file from the template media category index

7.1.109 createfolder

Syntax

createfolder (\$site, \$location, \$foldernew, \$user)

Input parameters

\$site ... publication name

\$location ... location

\$foldernew ... folder name

\$user ... user name

global input parameters

\$eventsystem

\$mgmt_config

\$cat

\$pageaccess

\$compaccess

\$hiddenfolder

\$hcms_linking

\$hcms_lang

\$lang

Output

result array

Description

This function creates a new folder. The folder name must not match the temp file patterns defined in include/tempfilepatterns.inc.php

7.1.110 createfolders

Syntax

createfolders (\$site, \$location, \$foldernew, \$user)

Input parameters

\$site ... publication name

\$location ... location

\$foldernew ... folder name

\$user ... user name

global input parameters

\$eventsystem

\$mgmt_config

\$cat

\$pageaccess

\$compaccess

\$hiddenfolder

\$hcms_linking

\$hcms_lang

\$lang

Output

array

Description

This function creates all folders recursively

7.1.111 collectfolders

Syntax

collectfolders (\$site, \$location, \$folder)

Input parameters

\$site ... publication name \$location ... location

\$folder ... folder name

global input parameters

\$user
\$mgmt_config
\$hcms_lang
\$lang

Output

result array / false

Description

Help function to create the collection of folders

7.1.112 copyfolders

Syntax

copyfolders (\$site, \$location, \$locationnew, \$folder, \$user)

Input parameters

\$site ... publication name \$location ... location (source) \$locationnew ... new location (destination) \$folder ... folder name \$user ... user name

global input parameters

\$mgmt_config \$cat \$pageaccess \$compaccess \$hiddenfolder \$hcms_linking \$hcms_lang \$lang

Output

result array equal to createfolder

Description

This function copies/creates all folders of the source location using mkdir (only directories will be created!). used for pasteobject function.

7.1.113 deletefolder

Syntax

deletefolder (\$site, \$location, \$folder, \$user)

Input parameters

\$site ... publication name \$location ... location \$folder ... folder name \$user ... user name

global input parameters

\$eventsystem \$mgmt_config \$cat \$pageaccess \$compaccess \$hiddenfolder \$hcms_linking \$hcms_lang \$lang

Output

array

Description

This function removes an folder. The folder must be empty in order to be removed from the system.

7.1.114 renamefolder

Syntax

renamefolder (\$site, \$location, \$folder, \$foldernew, \$user)

Input parameters

\$site ... publication name \$location ... location \$folder ... folder name \$foldernew ... new folder name \$user ... user name

global input parameters

\$eventsystem \$mgmt_config \$cat \$pageaccess \$compaccess \$hiddenfolder \$hcms_linking \$hcms_lang \$lang

Output

array

Description

This function renames a folder

7.1.115 correct container name

Syntax

correctcontainername (\$container_id)

Input parameters

\$container_id ... container ID

Output

corrected name / false on error

Description

This function adds zeros to the container ID to create the correct file name of the content container.

7.1.116 createobject

Syntax

createobject (\$site, \$location, \$page, \$template, \$user)

Input parameters

\$site ... publication name \$location ... location \$page ... object \$template ... template name \$user

global input parameters

\$eventsystem \$mgmt_config \$pageaccess \$compaccess \$hiddenfolder \$hcms_linking \$hcms_lang \$lang

Output

result array

Description

This function creates a new page or component

7.1.117 uploadfile

Syntax

uploadfile (\$site, \$location, \$cat, \$global_files, \$page="", \$unzip=0, \$createthumbnail=0, \$imageresize="", \$imagepercentage="", \$user="sys", \$checkduplicates=true, \$versioning=false)

Input parameters

\$site ... publication name
\$location ... destination location
\$cat ... category [page/comp]
\$global_files ... uploaded file (array as defined by PHP autoglobale \$_FILES)
\$page ... unzip [1/0]
\$unzip ... object name (only for media file update of existing object)
\$createthumbnail ... create only a new thumbnail [1/0]
\$imageresize ... imageresize [percentage
\$imagepercentage ... null]
\$user ... imagepercentage (%-value as integer)
\$checkduplicates ... user name
\$versioning ... check for duplicates [true, false]

global input parameters

\$mgmt_config \$mgmt_uncompress \$mgmt_imagepreview \$mgmt_mediapreview \$mgmt_mediapptions \$mgmt_imageoptions \$mgmt_maxsizepreview \$mgmt_parser \$eventsystem \$pageaccess \$compaccess \$hiddenfolder \$localpermission \$hcms_lang

Output

\$lang

result array

Description

This function manages all file uploads, like unzip files, create media objects and resize images.

The container name will be extracted from the media file name for updating an existing multimedia file.

7.1.118 createmediaobject

Syntax

createmediaobject (\$site, \$location, \$file, \$path_source_file, \$user, \$imagepercentage=0)

Input parameters

\$site ... publication name

\$location ... destination location

\$file ... file name

\$path_source_file ... path to source multimedia file (uploaded file in temp directory)

\$user ... user name

\$imagepercentage ... resize original image (100%) by percentage (optional)

global input parameters

\$mgmt_config

\$mgmt_imageoptions

\$eventsystem

\$pageaccess

\$compaccess

\$hiddenfolder

\$hcms_linking

\$hcms_lang

\$lang

Output

result array

Description

This function creates an asset (multimedia object) by reading a given source file. The file name must not match the temp file patterns defined in include/tempfilepatterns.inc.php and .recycle for recycled files.

The metadata template is based on the template of the folder the objects resides in.

7.1.119 createmediaobjects

Syntax

createmediaobjects (\$site, \$location_source, \$location_destination, \$user)

Input parameters

\$site ... publication name

\$location_source ... source location

\$location_destination ... destination location \$user ... user name

global input parameters

\$mgmt_config

\$mgmt_imageoptions

\$eventsystem

\$pageaccess

\$compaccess

\$hiddenfolder

\$hcms_linking

\$hcms_lang

\$lang

Output

result array with all objects created / false

Description

This function creates media objects by reading all media files from a given source location (used after unzipfile).

The file namey must not match the temp file patterns defined in include/tempfilepatterns.inc.php

7.1.120 editmediaobject

Syntax

editmediaobject (\$site, \$location, \$page, \$format="jpg", \$type="thumbnail", \$user)

Input parameters

\$site ... publication

\$location ... locationr

\$page ... object name

\$format ... format (file extension w/o dot) (optional)

\$type ... type of image/video/audio file [thumbnail, origthumb(thumbail made from original video/audio), original, any other string present in \$mgmt_imageoptions] (optional) \$user

global input parameters

\$wf_token

\$mgmt_config

\$mgmt_imagepreview

\$mgmt_mediapreview

\$mamt mediaoptions

\$mgmt_imageoptions

\$mgmt_maxsizepreview

\$mgmt_mediametadata

\$hcms_ext

\$hcms_lang

\$lang

Output

result array / false on error (saves original or thumbnail media file of an object for thumbnail only jpeg format is supported as output) user name

Description

This function mainly uses function createmedia to render the objects media, but at the same time takes care of versioning and the object name, if the file extension has been changed

7.1.121 manipulateobject

Syntax

manipulateobject (\$site, \$location, \$page, \$pagenew, \$user, \$action, \$clipboard_array=array())

Input parameters

\$site ... publication name
\$location ... location
\$page ... object name
\$pagenew ... new object name (exkl. extension except for action "file_rename")
\$user ... user
\$action ... action [page_delete
\$clipboard_array ... page_rename

global input parameters

\$wf_token \$eventsystem \$mgmt_config \$mgmt_mediaoptions \$mgmt_docoptions \$hcms_ext \$pageaccess \$compaccess \$hiddenfolder \$hcms_linking \$cat \$hcms_lang \$lang

Output

array

Description

This function removes, unpublishs, renames and pastes objects and is used by other functions which works as a shell for this function

7.1.122 deletemarkobject

Syntax

deletemarkobject (\$site, \$location, \$page, \$user)

Input parameters

\$site ... publication name \$location ... location \$page ... object \$user ... user name

global input parameters

\$wf_token \$eventsystem \$mgmt_config \$cat \$pageaccess \$compaccess \$hiddenfolder \$hcms_linking \$hcms_lang

\$lang

Output

result array

Description

This function marks a page, asset, or component as deleted.

7.1.123 deleteunmarkobject

Syntax

deleteunmarkobject (\$site, \$location, \$page, \$user)

Input parameters

\$site ... publication name \$location ... location \$page ... object \$user ... user name

global input parameters

\$wf_token \$eventsystem \$mgmt_config \$cat \$pageaccess \$compaccess \$hiddenfolder \$hcms_linking \$hcms_lang \$lang

Output

result array

Description

This function unmarks a page, asset, or component as deleted.

7.1.124 deleteobject

Syntax

deleteobject (\$site, \$location, \$page, \$user)

Input parameters

\$site ... publication name \$location ... location \$page ... object \$user ... user name

global input parameters

\$eventsystem \$mgmt_config \$cat \$pageaccess \$compaccess \$hiddenfolder \$hcms_linking \$hcms_lang \$lang

result array

Description

This function removes a page, asset, or component by calling the function manipulateobject.

7.1.125 renameobject

Syntax

renameobject (\$site, \$location, \$page, \$pagenew, \$user)

Input parameters

\$site ... publication name\$location ... location\$page ... object\$pagenew ... new object name exkl. file extension\$user ... user

global input parameters

\$eventsystem \$mgmt_config \$cat \$pageaccess \$compaccess \$hiddenfolder \$hcms_linking \$hcms_lang \$lang

Output

array

Description

This function renames a page or component and calls the function manipulateobject

7.1.126 renamefile

Syntax

renamefile (\$site, \$location, \$page, \$pagenew, \$user)

Input parameters

\$site ... publication name\$location ... location\$page ... object\$pagenew ... new object including file extension\$user ... user name

global input parameters

\$eventsystem \$mgmt_config \$cat \$pageaccess \$compaccess \$hiddenfolder \$hcms_linking \$hcms_lang \$lang

array

Description

This function renames a file (NOT a page or component) and calls the function manipulateobject.

This function renames the file name including the extension and not only the name of an object.

The event that will be executed in the event system is the same as renameobject.

7.1.127 cutobject

Syntax

cutobject (\$site, \$location, \$page, \$user, \$clipboard_add=false, \$clipboard_session=true)

Input parameters

\$site ... publication name[string]
\$location ... location[string]
\$page ... object[string]
\$user ... user[string]
\$clipboard_add ... add to existing clipboard entries [true, false] (optional)
\$clipboard_session ... save clipboard in session [true, false] (optional)

global input parameters

\$eventsystem
\$mgmt_config
\$cat
\$hcms_lang
\$lang

Output

array

Description

This function cuts a page or component

7.1.128 copyobject

Syntax

copyobject (\$site, \$location, \$page, \$user, \$clipboard_add=false, \$clipboard_session=true)

Input parameters

\$site ... publication name[string]
\$location ... location[string]
\$page ... object[string]
\$user ... user[string]
\$clipboard_add ... add to existing clipboard entries [true, false] (optional)
\$clipboard_session ... save clipboard in session [true, false] (optional)

global input parameters

\$eventsystem
\$mgmt_config
\$cat
\$hcms_lang
\$lang

Output

array

Description

This function copies a page or component

7.1.129 copyconnectedobject

Syntax

copyconnectedobject (\$site, \$location, \$page, \$user, \$clipboard_add=false, \$clipboard_session=true)

Input parameters

\$site ... publication name[string]
\$location ... location[string]
\$page ... object[string]
\$user ... user[string]
\$clipboard_add ... add to existing clipboard entries [true, false] (optional)
\$clipboard_session ... save clipboard in session [true, false] (optional)

global input parameters

\$eventsystem \$mgmt_config \$cat \$hcms_lang \$lang

Output

array

Description

This function makes a connected copy of a page or component

7.1.130 pasteobject

Syntax

pasteobject (\$site, \$location, \$user, \$clipboard_array=array())

Input parameters

\$site ... publication name[string]
\$location ... location[string]
\$user ... user[string]
\$clipboard_array ... clipboard entries as array (optional)

global input parameters

\$eventsystem \$mgmt_config \$cat \$pageaccess \$compaccess \$hiddenfolder \$hcms_linking \$hcms_lang \$lang

Output

array

Description

This function pastes an object by calling and calls the function manipulateobject

7.1.131 lockobject

Syntax

lockobject (\$site, \$location, \$page, \$user)

Input parameters

\$site ... publication name
\$location ... location
\$page ... object
\$user ... user

global input parameters

\$eventsystem
\$mgmt_config
\$cat
\$hcms_lang
\$lang

Output

array

Description

This function locks an object for a specific user

7.1.132 unlockobject

Syntax

unlockobject (\$site, \$location, \$page, \$user)

Input parameters

\$site ... publication name \$location ... location \$page ... object \$user ... user

global input parameters

\$eventsystem \$mgmt_config \$cat \$hcms_lang \$lang

Output

array

Description

This function unlocks an object of a specific user

7.1.133 publishobject

Syntax

publishobject (\$site, \$location, \$page, \$user)

Input parameters

\$site ... publication name

\$location ... location

\$page ... object (full name incl. extension)

\$user

global input parameters

\$eventsystem

\$mgmt_config

\$cat

\$ctrlreload

\$pageaccess

\$compaccess

\$hiddenfolder

\$hcms_linking

\$hcms_lang

\$lang

Output

array

Description

This function publishes a page, component or asset

7.1.134 publishlinkedobject

Syntax

publishlinkedobject (\$site, \$location, \$page, \$user)

Input parameters

\$site ... publication name \$location ... location \$page ... object \$user ... user name

global input parameters

\$eventsystem \$mgmt_config \$cat \$ctrlreload \$hcms_lang \$lang

Output

array

Description

This function publishes all linked objects of a given object.

All objects with component links (references) to the given object will be published.

This funtion is only used by publishobject.

7.1.135 unpublishobject

Syntax

unpublishobject (\$site, \$location, \$page, \$user)

Input parameters

\$site ... publication name \$location ... location \$page ... object

\$user

global input parameters

\$eventsystem

\$mgmt_config

\$cat

\$ctrlreload

\$pageaccess

\$compaccess

\$hiddenfolder

\$hcms_linking

\$hcms_lang

\$lang

Output

array

Description

This function unpublishes a page or component and calls the function manipulateobject

7.1.136 processobjects

Syntax

processobjects (\$action, \$site, \$location, \$file, \$published_only="0", \$user)

Input parameters

\$action ... action [publish, unpublish, delete]

\$site ... publication name

\$location ... location

\$file ... object name

\$published_only ... only published objects [pub, all]

\$user ... user name

global input parameters

\$eventsystem

\$mgmt_config

\$pageaccess

\$compaccess

\$hiddenfolder

\$hcms_linking

\$hcms_lang

\$lang

Output

true/false on error

Description

Publish, unpublish or delete all objects recursively. This function is used by the job 'minutely' to process all objects of the queue.

In order to process all objects recursively a folder name need to be provided and not the .folder file.

This function should not be used for the graphical user interface since it does not provide feedback about the process state!

7.1.137 collectobjects

Syntax

collectobjects (\$root_id, \$site, \$cat, \$location, \$published_only="0")

Input parameters

\$root_id ... root ID

\$site ... publication name

\$cat ... category [page, comp]

\$location ... location

\$published_only ... collect only published objects [0, 1]

global input parameters

\$user

\$pageaccess

\$compaccess

\$mgmt_config

\$hiddenfolder

\$hcms_lang

\$lang

Output

result array / false

Description

Help function used to create a list of all objects inside a folder

7.1.138 manipulateallobjects

Syntax

manipulateallobjects (\$action, \$objectpath_array, \$method="", \$force="start", \$published_only=0, \$user, \$tempfile="", \$maxitems=5)

Input parameters

\$action ... action [publish

\$objectpath_array ... unpublish

\$method ... deletemark

\$force ... deleteunmark/restore

\$published_only ... emptypin

\$user ... delete

\$tempfile ... paste]

\$maxitems ... objectpath (array)

global input parameters

\$eventsystem

\$mgmt_config

\$cat

\$pageaccess

\$compaccess

\$hiddenfolder

\$hcms_lang

\$lang

Output

true/false

Description

This function is used to perform actions on multiple objects and is mainly used by popup_status.php.

This functions should only be used in connection with the GUI of the system.

7.1.139 remoteclient

Syntax

remoteclient (\$action, \$root, \$site, \$location, \$locationnew, \$page, \$pagenew)

Input parameters

\$action ... action [save \$root ... copy \$site ... delete \$location ... rename \$locationnew ... get] \$page ... root [abs_path_link \$pagenew ... abs_path_media

global input parameters

\$user
\$mgmt_config
\$hcms_lang
\$lang

Output

http answer [string] or false

Description

Sends data to remote client via http post

7.1.140 HTTP_Post

Syntax

HTTP_Post (\$URL, \$data, \$contenttype="application/x-www-form-urlencoded", \$charset="UTF-8", \$referrer="")

Input parameters

\$URL ... URL[string]
\$data ... \$data[array] (raw data)
\$contenttype ... content-type [application/x-www-form-urlencoded
\$charset ... multipart/form-data]
\$referrer ... character set [string]

Output

http response [string] / false on error

Description

Sends data via http post and returns response

7.1.141 HTTP_Get

Syntax

HTTP_Get (\$URL, \$data="", \$contenttype="application/x-www-form-urlencoded", \$charset="UTF-8")

Input parameters

\$URL ... URL[string]
\$data ... \$data[array] (raw data) (optional)
\$contenttype ... content-type[string excl. charset] (optional)
\$charset ... character set[string] (optional)

Output

http response [string] / false on error

Description

Sends data via http get and returns response

7.1.142 HTTP_Proxy

Syntax

HTTP_Proxy (\$URL, \$enable_file=false)

Input parameters

\$URL ... URL[string]

\$enable_file ... enable post of files [true, false] (optional)

global input parameters

\$mgmt_config

Output

http response [string] / false on error

Description

Sends all global POST/GET and FILES data via http post and returns response

7.1.143 loadbalancer

Syntax

loadbalancer (\$type)

Input parameters

\$type ... type [renderimage, rendervideo, uploadfile]

global input parameters

\$mgmt_config

Output

http response [string] / false on error or if disabled

Description

Balances the load by sending all global POST/GET and FILES to one service ressource of a given array of service ressources.

Don't define and use the same server ressources in \$mgmt_config['url_path_service'], this can lead to an infinite loop.

7.1.144 savelog

Syntax

savelog (\$error, \$logfile="event")

Input parameters

\$error ... name of log file without extension (optional)
\$logfile ... return_type [string, array] (optional)

global input parameters

\$user \$eventsystem \$mgmt_config \$hcms_lang \$lang

true / false on error

Description

Loads a log file an returns the data as string or array for all log records.

7.1.145 deletelog

Syntax

deletelog (\$logname="")

Input parameters

\$logname ... logname (optional)

global input parameters

\$user \$eventsystem \$mgmt_config \$hcms_lang \$lang

Output

result array

Description

Deletes a log file.

7.1.146 debuglog

Syntax

debuglog (\$code)

Input parameters

\$code ... code to write to debug file

global input parameters

\$user
\$mgmt_config
\$hcms_lang
\$lang

Output

true / false

Description

Writes code lines into debug file in data/log/debug.log

7.1.147 notifyusers

Syntax

notifyusers (\$site, \$location, \$object, \$event, \$user_from)

Input parameters

\$site ... publication name\$location ... location\$object ... object name\$event ... event name [oncreate, onedit, onmove, ondelete]

\$user_from ... user name

global input parameters

\$user
\$mgmt_config
\$hcms_lang_codepage
\$hcms_lang
\$lang

Output

true / false on error

Description

Notifies all users based on the given event and location

7.1.148 sendlicensenotification

Syntax

sendlicensenotification (\$site, \$cat, \$folderpath, \$text_id, \$date_begin, \$date_end, \$user, \$format="%Y-%m-%d")

Input parameters

\$site ... publication name
\$cat ... category [page, comp]
\$folderpath ... folder path
\$text_id ... text ID for text field
\$date_begin ... search from date (YYYY-MM-DD)
\$date_end ... search till date (YYYY-MM-DD)
\$user ... user name string or array (optional)
\$format ... date format (optional)

global input parameters

\$eventsystem \$mgmt_config \$hcms_lang_codepage \$hcms_lang \$lang

Output

true / false on error

Description

Searches for objects with a date in a defined text field that has to be between the defined date limits and sends a message to the defined users.

This is a helper function for function licensenotification.

7.1.149 licensenotification

Syntax

licensenotification ()

Input parameters

global input parameters

\$eventsystem
\$mgmt_config
\$hcms_lang_codepage
\$hcms_lang

\$lang

Output

true / false on error

Description

This function reads the license notification configuration and looks up all objects with a date in a defined text field

that has to be between the defined date limits and sends a message to the defined users.

7.1.150 html_diff

Syntax

html_diff (\$old, \$new)

Input parameters

\$old ... old text \$new ... new text

Output

result text showing deleted and inserted words/differences / false on error

Description

Paul's Simple Diff Algorithm v 0.1

Function html_diff is a wrapper for the diff command, it takes two strings and returns the differences in HTML. The tags used are <ins> and , which can easily be styled with CSS.

7.1.151 createfavorite

Syntax

createfavorite (\$site="", \$location="", \$page="", \$id="", \$user)

Input parameters

\$site ... publication name (optional) \$location ... location (optional) \$page ... object name (optional) \$id ... identifier (object ID \$user ... object hash) (optional)

global input parameters

\$mgmt_config

Output

true / false

7.1.152 deletefavorite

Syntax

deletefavorite (\$site="", \$location="", \$page="", \$id="", \$user)

Input parameters

\$site ... publication name (optional) \$location ... location (optional) \$page ... object name (optional) \$id ... identifier (object ID \$user ... object hash) (optional)

global input parameters

\$mgmt_config

Output

true / false

7.1.153 rewrite_targetURI

Syntax

rewrite_targetURI (\$site, \$text_id, \$uri, \$exclude_dir_esc="", \$rewrite_type="include")

Input parameters

\$site ... publication name
\$text_id ... text ID array (text-ID as key and URL paramaters as value)
\$uri ... requested URI as string
\$exclude_dir_esc ... exclude path as array (optional)
\$rewrite_type ... rewrite type [none, forward, include] (optional)

global input parameters

\$mgmt_config
\$publ_config

Output

target URI / false on error

7.1.154 rewrite_homepage

Syntax

rewrite_homepage (\$site, \$rewrite_type="forward")

Input parameters

\$site ... publication name
\$rewrite_type ... rewrite type [none, forward] (optional)

global input parameters

\$mgmt_config
\$publ_config

Output

target URI / false on error

Description

Uses the page root directory of the publication configuration and forwards to the default index page. No page include supported!

7.1.155 sendmessage

Syntax

sendmessage (\$from_user, \$to_user, \$title, \$message, \$object_id="")

Input parameters

\$from_user ... from_user name \$to_user ... to_user name \$title ... title \$message ... message \$object_id ... object ID or object path (optional)

global input parameters

\$mgmt_config
\$hcms_lang_codepage
\$hcms_lang
\$lang

Output

true/false

Description

Sends a message via e-mail to a user

7.2 Get API Functions

7.2.1 getserverload

Syntax

getserverload ()

Input parameters

Output

Returns the average system load (the number of processes in the system run queue) over the last minute and the number of CPU cores as array

7.2.2 getconfigvalue

Syntax

getconfigvalue (\$config, \$in_key="")

Input parameters

\$config ... settings array

\$in_key ... value/substring in array key (optional)

Output

value of setting

Description

Help function for createinstance

7.2.3 getsession

Syntax

getsession (\$variable, \$default="")

Input parameters

\$variable ... session variable name
\$default ... default session value (optional)

Output

session value

7.2.4 getrequest

Syntax

getrequest (\$variable, \$force_type=false, \$default="")

Input parameters

\$variable ... request variable name

\$force_type ... must be of certain type [numeric, array, publicationname, locationname, objectname, url, bool] (optional)

\$default ... default value (optional)

Output

request value

Description

Returns the value of a POST, GET or COOKIE request, or a default value if not valid.

7.2.5 getrequest_esc

Syntax

getrequest_esc (\$variable, \$force_type=false, \$default="", \$js_protection=false)

Input parameters

\$variable ... request variable name

\$force_type ... must be of certain type [numeric, array, publicationname, locationname, objectname] (optional)

\$default ... default value (optional)

\$js_protection ... remove characters to avoid JS injection [true, false] (optional)

Output

request value

Description

Returns the escaped value in order to prevent XSS from POST, GET or COOKIE variables. Returns a default value if not valid.

7.2.6 getuserip

Syntax

getuserip ()

Input parameters

Output

IP address of client / false on error

Description

Retrieves the IP address of the client/user.

7.2.7 getobjectlistcells

Syntax

getobjectlistcells (\$viewportwidth, \$is_mobile=0)

Input parameters

\$viewportwidth ... width of viewport or window in pixels \$is_mobile ... is mobile device [0, 1] (optional)

Output

number of table cells/rows for the gallery view of object lists

7.2.8 getlanguageoptions

Syntax

getlanguageoptions ()

Input parameters

global input parameters

\$mgmt_config

Output

array with 2-digit language code as key and language name in English as value / false on error

7.2.9 getlanguagefile

Syntax

getlanguagefile (\$lang="en")

Input parameters

\$lang ... language code (optional)

global input parameters

\$mgmt_config

Output

language file name

7.2.10 getcodepage

Syntax

getcodepage (\$lang="en")

Input parameters

\$lang ... language code (optional)

global input parameters

\$mgmt_config

\$hcms_lang_codepage

Output

code page (character set)

7.2.11 getcalendarlang

Syntax

getcalendarlang (\$lang="en")

Input parameters

\$lang ... language code (optional)

global input parameters

\$mgmt_config

Output

supported language code for calendar

7.2.12 getescapedtext

Syntax

getescapedtext (\$text, \$charset="", \$lang="")

Input parameters

\$text ... word as string\$charset ... 2-digit language code (optional)\$lang

global input parameters

\$mgmt_config

Output

array holding all synonyms including the provided word / false on error

Description

Returns the synonyms of a word.

7.2.13 getsearchhistory

Syntax

getsearchhistory (\$user="")

Input parameters

\$user ... user name (optional)

global input parameters

\$mgmt_config

Output

array holding all expressions (in single quotes) of the search history of a user / false on error

7.2.14 gettaxonomy_sublevel

Syntax

gettaxonomy_sublevel (\$site, \$lang="en", \$tax_id="0")

Input parameters

\$site ... publication name
\$lang ... language code (optional)
\$tax_id ... taxonomy parent ID (optional)

global input parameters

\$mgmt_config
\$mgmt_lang_shortcut_default
\$taxonomy

Output

array holding all keywords of the next taxonomy level / false on error

Description

Returns sorted keywords of a taxonomy level (multilingual support based on taxonomies). Global variable \$taxonomy can be used to pass the taxonomy as array.

7.2.15 gettaxonomy_childs

Syntax

gettaxonomy_childs (\$site="", \$lang="", \$expression, \$childlevels=1, \$id_only=true)

Input parameters

\$site ... publication name (optional)

\$lang ... language code (optional)

\$expression ... taxonomy ID or expression or taxonomy path in the form

%taxonomy%/publication-name or 'default'/language-code/taxonomy-ID/taxonomy-child-levels as string

\$childlevels ... taxonomy child levels as integer (optional)

\$id_only ... only return taxonomy IDs without language and keyword information [true, false] (optional)

global input parameters

\$mgmt_config
\$mgmt_lang_shortcut_default
\$taxonomy

Output

array holding all taxonomy IDs / false on error

Description

Returns keywords based on taxonomy defintion and synonyms if expression is a keyword (multilingual support based on taxonomies and synonyms).

The expression can be a taxonomy path in the form of %taxonomy%/site/language-code/taxonomy-ID/taxonomy-child-levels (use "all" for all languages and "0" for all taxonomy-IDs on first level).

Global variable \$taxonomy can be used to pass the taxonomy as array.

7.2.16 gethierarchy_defintion

Syntax

gethierarchy_defintion (\$site, \$selectname="")

Input parameters

\$site ... publication name

\$selectname ... hierarchy name (optional)

global input parameters

\$mgmt_config

Output

hierarchy array in form of array[name][level][text-id][language] = label / false on error

Description

Reads the metadata/content hierarchy defintion and returns a multidimensinal array.

7.2.17 gethierarchy_sublevel

Syntax

gethierarchy_sublevel (\$hierarchy_url)

Input parameters

\$hierarchy_url ... hierarchy URL in form of %hierarchy%/publication-name/hierarchy-name/hierarchy-level-of-last-element/text-ID-1=value-1/text-ID-2=value-2/text-ID-3

global input parameters

\$mgmt_config
\$lang

Output

array holding all hierarchy URLs as key and text content or label as value / false on error

Description

Returns sorted values of a metadata/content hierarchy level.

7.2.18 getkeywords

Syntax

getkeywords (\$site="")

Input parameters

\$site ... publication name (optional)

global input parameters

\$mgmt_config

Output

keywords as array / false on error

Description

Generates an array holding all keywords and the number as value and keyword ID as key.

7.2.19 getmetakeywords

Syntax

getmetakeywords (\$text, \$language="en", \$charset="UTF-8")

Input parameters

\$text ... text as string \$language ... language to be used for stop word list [de, en, ...] (optional) \$charset ... character set (optional)

global input parameters

\$mgmt_config

Output

keywords as array /false on error

Description

Generates a keyword list from a plain text. Stop word lists are defined in data/include/stopwords.inc.php

7.2.20 getmetadescription

Syntax

getmetadescription (\$text, \$charset="UTF-8")

Input parameters

\$text ... text as string \$charset

Output

cleanded description of provided text /false on error

Description

Generates a description from a text, to be used as meta information.

7.2.21 getgooglesitemap

Syntax

getgooglesitemap (\$site, \$dir, \$url, \$getpara=array(), \$permalink=array(), \$chfreq="weekly", \$prio="", \$ignore=array(), \$filetypes=array('cfm', 'htm', 'html', 'xhtml', 'asp', 'aspx', 'jsp', 'php', 'pdf'), \$show_freq=true, \$show_prio=true)

Input parameters

\$site ... publication anme
\$dir ... directory path
\$url ... URL to directory
\$getpara ... GET parameters to use for new versions of the URL as array (optional)
\$permalink ... permanent links text-ID to use for location as array (optional)
\$chfreq ... frequency of google scrawler [never, weekly, daily] (optional)
\$prio ... priority [1 or less] (optional)
\$ignore ... ignore file names as array (optional)
\$filetypes ... allowed file types as array (optional)
\$show_freq ... include frequency tag [true, false] (optional)
\$show_prio ... include priority tag [true, false] (optional)

global input parameters

\$mgmt_config
\$publ_config

Output

xml sitemap / false on error

Description

Generates a google sitemap xml-output

7.2.22 getlistelements

Syntax

getlistelements (\$list_sourcefile)

Input parameters

\$list_sourcefile ... content of file attribute of list of keyword tag

global input parameters

\$mgmt_config
\$lang

Output

string with list/keyword elements sperated by commas / false

7.2.23 getmetadata

Syntax

getmetadata (\$location, \$object, \$container="", \$seperator="\n", \$template="")

Input parameters

\$location ... location

\$object ... object (both optional if container is given)

\$container ... container name/ID or container content (optional)

\$seperator ... seperator of meta data fields [any string, array] (optional)
\$template ... publication name/template name to extract label names (optional)

global input parameters

\$mgmt_config

Output

string with all metadata from given object based on container / false

7.2.24 getmetadata_multiobjects

Syntax

getmetadata_multiobjects (\$multiobject_array, \$user)

Input parameters

\$multiobject_array ... converted path of multiple objects as array \$user ... user name

global input parameters

\$mgmt_config \$siteaccess \$pageaccess \$compaccess \$hiddenfolder

\$adminpermission

\$localpermission

Output

assoziatve array with all text content and meta data / false

Description

Extracts all metadata including media information for a provided list of objects. This function is used for the CSV export in the objectlist views and also evaluates the access permissions of the user.

7.2.25 getmetadata_container

Syntax

getmetadata_container (\$container_id, \$text_id_array)

Input parameters

\$container_id ... container ID
\$text_id_array ... array of text IDs

global input parameters

\$mgmt_config \$labels

Output

assoziatve array with all text content and meta data / false

Description

Extracts container, media, and metadata information of a container. This function is used for the presentation of metadata for objectlist views.

7.2.26 getobjectcontainer

Syntax

getobjectcontainer (\$site, \$location, \$object, \$user)

Input parameters

\$site ... publication [string] \$location ... location [string] \$object ... object [string] \$user ... user [string]

global input parameters

\$mgmt_config

Output

Content Container [XML]/false

Description

Loads the content container of a given object (page, component, folder)

7.2.27 getcontainer

Syntax

getcontainer (\$containerid, \$type)

Input parameters

\$containerid ... container name or container ID \$type ... container type [published

global input parameters

\$mgmt_config

Output

Contant Container [XML]/false

Description

Obsolete function used as an alias for the loadcontainer function without the possibility to load locked containers

7.2.28 getwallpaper

Syntax

getwallpaper (\$version="")

Input parameters

\$version ... version number (optional)

global input parameters

\$mgmt_config

Output

URL of wallpaper image / false

Description

Provides a wallpaper image or video used for the logon and home screen.

7.2.29 getcontainername

Syntax

getcontainername (\$container)

Input parameters

\$container ... container name (e.g. 0000112.xml.wrk) or container ID

global input parameters

\$mgmt_config

Output

Array with file name of the working content container (locked or unlocked!) and username if locked

7.2.30 getlocationname

Syntax

getlocationname (\$site, \$location, \$cat, \$source="path")

Input parameters

\$site ... publication name
\$location ... location path (as absolute path or converted path)
\$cat ... category [page, comp]
\$source ... source for name [path, name]

global input parameters

\$mgmt_config \$lang \$hcms_lang_codepage

Output

location with readable names instead of file names / false on error

7.2.31 getthemelocation

Syntax

getthemelocation (\$theme="")

Input parameters

\$theme ... theme name (optional)

global input parameters

\$mgmt_config

Output

path to theme / false

Description

Returns the absolute path (URL) of the theme (css and images).

7.2.32 getcategory

Syntax

getcategory (\$site="", \$location)

Input parameters

\$site ... publication name (optional) \$location ... location path

global input parameters

\$mgmt_config
\$publ_config

category ['page comp'] / false on error

Description

Evaluates the category ['page, comp'] of a location

7.2.33 getpublication

Syntax

getpublication (\$path)

Input parameters

\$path ... converted location path

Output

publication name

Description

Extract the publication name of a location path

7.2.34 getlocation

Syntax

getlocation (\$path)

Input parameters

\$path ... location path

Output

location (without object or folder)

Description

Extract the location excluding object or folder of a location path

7.2.35 getobject

Syntax

getobject (\$path)

Input parameters

\$path ... location path

Output

object or folder name

Description

Extract the object or folder of a location path

7.2.36 getmediacontainername

Syntax

getmediacontainername (\$file)

Input parameters

\$file ... file name

container name / false on error

Description

Extract the container name from a multimedia file name by using the hcm-ID

7.2.37 getmediafileversion

Syntax

getmediafileversion (\$container)

Input parameters

\$container ... container name or container ID

global input parameters

\$mgmt_config
\$user

Output

media file name / false on error

Description

Extracts the name from the multimedia file by container name or ID in order to get the media file of older content versions.

if the result is false, there is no older media file version.

7.2.38 getobjectid

Syntax

getobjectid (\$objectlink)

Input parameters

\$objectlink ... converted object path or pathes separated by |

Output

object ID

Description

Converts the object path to the object ID of any object

7.2.39 getobjectlink

Syntax

getobjectlink (\$objectid)

Input parameters

\$objectid ... converted object ID or IDs separated by |

Output

converted object link

Description

Converts the object ID to the object path of any object

7.2.40 getcontainerversions

Syntax

getcontainerversions (\$container)

Input parameters

\$container ... container ID or container name

global input parameters

\$mgmt_config

Output

array of all versions (array[version-extension] = file-name) / false

7.2.41 getlocaltemplates

Syntax

getlocaltemplates (\$site, \$cat="")

Input parameters

\$site ... publication name

\$cat ... template category [page, comp, meta, inc] (optional)

global input parameters

\$mgmt_config

Output

array with all template names / false

Description

This function returns a list of all templates of a publication without inherited templates from other publications.

7.2.42 gettemplates

Syntax

gettemplates (\$site, \$cat="all")

Input parameters

\$site ... publication name

\$cat ... object category [all, page, comp, meta] (optional)

global input parameters

\$user \$mgmt_config \$hcms_lang \$lang

Output

template file name list as array / false on error

Description

This function returns a list of all templates for pages or components.

Based on the inheritance settings of the publication the template will be loaded with highest priority from the own publication and if not available from a parent publication.

7.2.43 gettemplateversions

Syntax

gettemplateversions (\$site, \$template)

Input parameters

\$site ... publication name \$template ... template name

global input parameters

\$mgmt_config

Output

array of all versions (array['YYYY-MM-DD HH:MM:SS'] = file-name) / false

7.2.44 getfileinfo

Syntax

getfileinfo (\$site, \$file, \$cat="comp")

Input parameters

\$site ... publication name (optional)
\$file ... file name incl. extension
\$cat ... category [page, comp] (optional)

global input parameters

\$mgmt_config

Output

array/false

Description

defines file properties based on the file extension and returns file info as an array:

\$result['file']: file name without hypercms management extension

\$result['name']: readable file name without hypercms management extension

\$result['filename']: file name without file extensions

\$result['icon']: file name of the file icon

\$result['type']: file type

\$result['ext']: file extension incl. dot in lower case

\$result['published']: if file is published = true else = false \$result['deleted']: if file is deleted = true else = false

7.2.45 getobjectinfo

Syntax

getobjectinfo (\$site, \$location, \$object, \$user="sys", \$container_version="")

Input parameters

\$site ... publication name
\$location ... location
\$object ... object name
\$user ... user name (optional)
\$container_version ... container version (optional)

global input parameters

\$mgmt_config

Output

result array / false on error

Description

Get all file pointers (container, media, template) and object name from object file and collect info from container version, if provided

7.2.46 getfilesize

Syntax

getfilesize (\$file)

Input parameters

\$file ... converted path to file or directory

global input parameters

\$mgmt_config

Output

result array with file size in kB and file count / false on error

Description

This function won't give you a proper result of the file size of multimedia components, if there is no Database installed.

7.2.47 getmimetype

Syntax

getmimetype (\$file)

Input parameters

\$file ... file name incl. extension

global input parameters

\$mgmt_config

Output

mime_type

Description

Gets the mime-type of the file by its extension.

If file has a version file extension the next file extension will be used.

7.2.48 getfiletype

Syntax

getfiletype (\$file_ext)

Input parameters

\$file_ext ... file extension or file name

global input parameters

\$mgmt_config
\$hcms_ext

Output

file type to be saved in database based on file extension

7.2.49 getpreviewwidth

Syntax

getpreviewwidth (\$site, \$filepath, \$width_orig="")

Input parameters

\$site ... publication name (optional)
\$filepath ... path to file or file name
\$width_orig ... original width (optional)

global input parameters

\$mgmt_config
\$hcms_ext

Output

result array with width and height / false on error

Description

Returns the default preview/annotation width in pixel of a document, image, or video

7.2.50 getpdfinfo

Syntax

getpdfinfo (\$filepath, \$box="MediaBox")

Input parameters

\$filepath ... path to PDF file \$box ... box attribute [BleedBox, CropBox, MediaBox] (optional)

global input parameters

\$mgmt_config \$mgmt_imagepreview \$user

Output

result array with width and height / false on error

Description

Extracts width and height in pixel of a PDF file based on the MediaBox in the filex content or ImageMagick as fallback

7.2.51 getvideoinfo

Syntax

getvideoinfo (\$mediafile)

Input parameters

\$mediafile ... path to video file

global input parameters

\$mgmt_config \$mgmt_mediapreview \$user

Output

video file information as result array / false on error

Description

Extract video metadata from video file.

7.2.52 getbrowserinfo

Syntax

getbrowserinfo ()

Input parameters

Output

client browser and version as array / false on error

7.2.53 getcontentlocation

Syntax

getcontentlocation (\$container_id, \$type="abs_path_content")

Input parameters

\$container_id ... container id
\$type ... type [url_path_content

global input parameters

\$mgmt_config

Output

location of the container file / false on error

Description

Gets the content location based on the given container id.

The use of various directories is necessary since the number of directories is limited by the filesystem, e.g. Linux ext3 is limited to 32000.

7.2.54 getmedialocation

Syntax

getmedialocation (\$site, \$file, \$type)

Input parameters

\$site ... publication name

 $file \dots multimedia file name (including hcm-ID)$

\$type ... type [url_path_media

global input parameters

\$mgmt_config
\$publ_config

Output

location of the multimedia file / false on error

Description

Gets the media repsitory location from \$mgmt_config array.

The function supports up to 10 media repositories.

Any other rules for splitting the media files on multiple devices can be implemented as well by the function getmedialocation_rule.

7.2.55 getlockedfileinfo

Syntax

getlockedfileinfo (\$location, \$file)

Input parameters

\$location ... location to file \$file ... file name

global input parameters

\$mgmt_config

Output

Array holding file name incl. lock extension and user name / false on error

Description

Finds the locked file and returns the name and user as array

7.2.56 getfavorites

Syntax

getfavorites (\$user, \$output="path")

Input parameters

\$user ... user name
\$output ... output [path, id] (optional)

global input parameters

\$mgmt_config

Output

object info or object id array of users favorites / false

7.2.57 getboxes

Syntax

getboxes (\$user)

Input parameters

\$user ... user name

global input parameters

\$mgmt_config

Output

selected home box names of user as array / false

7.2.58 getusersonline

Syntax

getusersonline ()

Input parameters

global input parameters

\$mgmt_config
\$siteaccess

Array of online user names / false

7.2.59 getchatstate

Syntax

getchatstate (\$register=true)

Input parameters

\$register ... register stat in session [true/false] (optional)

global input parameters

\$mgmt_config

Output

state of chat / false on error

7.2.60 getimagelib

Syntax

getimagelib ()

Input parameters

global input parameters

\$mgmt_imagepreview

Output

name of image library used [GD ImageMagick] / false on error

7.2.61 getfilename

Syntax

getfilename (\$filedata, \$tagname)

Input parameters

\$filedata ... file content

\$tagname ... hyperCMS tag name in page or component

Output

file name

Description

Extracts the file name of the content and template pointer tags of an object file

7.2.62 gethypertag

Syntax

gethypertag (\$filedata, \$tagname, \$offset=0)

Input parameters

\$filedata ... file content [string]

\$tagname ... full/partly hyperCMS tag name (with or without hyperCMS:) [string]

\$offset ... offset position [integer]

full hyperCMS tag array [array]/false on error

Description

Finds the hyperCMS tag start and end position and returns an array of the whole tags including all information.

Offset value must be integer value and is used to skip search for hyperCMS tag till offset position of filedata.

7.2.63 gethypertagname

Syntax

gethypertagname (\$tagdata)

Input parameters

\$tagdata ... full hyperCMS tag

Output

full hyperCMS tag name/false on error

Description

Reads the name of the hyperCMS tag

7.2.64 gethtmltag

Syntax

gethtmltag (\$filedata, \$tag)

Input parameters

\$filedata ... file content

\$tag ... full hyperCMS tag (or other identifier)

Output

full html tag/false on error

Description

Finds the first html tag start and end position of a nested hyperCMS tag and returns the whole tag including all information.

Works also if other script tags are nested in the HTML-tag.

This function is not case sensitive!

7.2.65 gethtmltags

Syntax

gethtmltags (\$filedata, \$tag)

Input parameters

\$filedata ... file content

\$tag ... full hyperCMS tag or other identifier in html tag

Output

string from html tag start to end tag/false on error

Description

Finds the nearest html tag start and end position of a nested hyperCMS tag and returns the whole tag including all information.

This functions works also for html-tag pairs like <a href>, <div></div> and so on.

7.2.66 getattribute

Syntax

getattribute (\$string, \$attribute, \$secure=true)

Input parameters

\$string ... string including attributes

\$attribute ... attribute name

\$secure ... secure attribute value reg. XSS (optional)

Output

attribute value/false on error

Description

Get the value of a certain attribute out of a string (...attributname=value....)

7.2.67 getoption

Syntax

getoption (\$string, \$option)

Input parameters

\$string ... string including options

\$option ... option name

Output

option value / false on error

Description

Get the value of a certain option out of a string (-c:v value -ar 44100)

7.2.68 getcharset

Syntax

getcharset (\$site, \$data)

Input parameters

\$site ... publication

\$data ... data from template or content container [string]

global input parameters

\$mgmt_config

Output

array with content-type and charset / false on error

Description

Extract the content-type definition and the character set from the template (1st priority), content container (2nd priority) or publication settings (3rd priority)

7.2.69 getartid

Syntax

getartid (\$id)

Input parameters

\$id ... string including id

article id/false on error

Description

Extract the article ID from the tag ID

7.2.70 getelementid

Syntax

getelementid (\$id)

Input parameters

\$id ... string including id

Output

element id/false on error

Description

Extract the element ID from the tag ID

7.2.71 getfirstkey

Syntax

getfirstkey (\$array)

Input parameters

\$array ... array

Output

array key of first element in array if \$value is not empty / false on error

7.2.72 getdirectoryfiles

Syntax

getdirectoryfiles (\$dir, \$pattern="")

Input parameters

\$dir ... path to directory

\$pattern ... pattern as string (optional)

Output

sorted array of all files matching the pattern / false on error

7.2.73 getuserinformation

Syntax

getuserinformation ()

Input parameters

global input parameters

\$mgmt_config
\$user

Output

assoziative array with basic user information [publication->username->attribute] / false

Description

This function creates an assoziative array with user information, e.g. for a user select box.

7.2.74 getworkflowitem

Syntax

getworkflowitem (\$site, \$workflow_file, \$workflow, \$user)

Input parameters

\$site ... publication name [string] \$workflow_file ... location name [string] \$workflow ... object name [string] \$user ... workflow file name [string]

global input parameters

\$mgmt_config
\$hcms_lang
\$lang

Output

workflow item [XML-string]

7.3 Set API Functions

7.3.1 setsession

Syntax

setsession (\$variable, \$content="", \$write=false)

Input parameters

\$variable ... temporary hyperCMS variable name or array\$content ... value as string or array (optional)\$write ... write session data for load balancer [true, false] (optional)

Output

true / false on error

7.3.2 settaxonomy

Syntax

settaxonomy (\$site, \$container_id, \$langcode="")

Input parameters

\$site ... publication name
\$container_id ... container ID
\$langcode ... 2-digit language code (optional)

global input parameters

\$mgmt_config \$taxonomy

Output

result array / false on error

Description

Analyzes the content regarding all taxonomy keywords, saves results in database and returns

an array (multilingual support based on taxonomies). Global variable \$taxonomy can be used to pass the taxonomy as array.

7.3.3 setarticle

Syntax

setarticle (\$site, \$contentdata, \$contentfile, \$arttitle, \$artstatus, \$artdatefrom, \$artdateto, \$artuser, \$user)

Input parameters

\$site ... publication name \$contentdata ... container (XML) \$contentfile ... container name \$arttitle ... article title array \$artstatus ... article status array \$artdatefrom ... article beginn date array \$artdateto ... article end date array \$artuser ... user array or string \$user ... user name

global input parameters

\$mgmt_config

Output

updated content container (XML) false on error

Description

Set article content in container. The content container will be returned and not saved.

7.3.4 settext

Syntax

settext (\$site, \$contentdata, \$contentfile, \$text, \$type, \$art, \$textuser, \$user, \$charset="", \$addmicrotime=false)

Input parameters

\$site ... publication name
\$contentdata ... container (XML)
\$contentfile ... container name
\$text ... text array
\$type ... type array or string [u, f, l, c, d, k]
\$art ... article array or string [yes, no]
\$textuser ... text user array or string
\$user ... user name
\$charset ... character set of text content
\$addmicrotime ... add microtime to ID [true, false] used for comments

global input parameters

\$mgmt_config \$publ_config

Output

updated content container (XML) false on error

Description

Set text content in container and database. The content container will be returned and not saved.

7.3.5 setmedia

Syntax

setmedia (\$site, \$contentdata, \$contentfile, \$mediafile, \$mediaobject_curr, \$mediaobject, \$mediaalttext, \$mediaalign, \$mediawidth, \$mediaheight, \$art, \$mediauser, \$user, \$charset="")

Input parameters

\$site ... publication name

\$contentdata ... container (XML)

\$contentfile ... container name

\$mediafile ... media arrays (some are optional)

\$mediaobject_curr ... article array or string [yes, no]

\$mediaobject ... content user array or string

\$mediaalttext ... user name

\$mediaalign ... chracter set of text content

\$mediawidth

\$mediaheight

\$art

\$mediauser

\$user

\$charset

global input parameters

\$mgmt_config

Output

updated content container (XML)

false on error

Description

Set media content in container and database. The content container will be returned and not saved.

7.3.6 setpagelink

Syntax

setpagelink (\$site, \$contentdata, \$contentfile, \$linkhref_curr, \$linkhref, \$linktarget, \$linktext, \$art, \$linkuser, \$user, \$charset="")

Input parameters

\$site ... publication name

\$contentdata ... container (XML)

\$contentfile ... container name

\$linkhref_curr ... current link array

\$linkhref ... new link array

\$linktarget ... link target array

\$linktext ... link text array

\$art ... article array or string [yes, no]

\$linkuser ... content user array or string

\$user ... user name

\$charset ... chracter set of text content

global input parameters

\$mgmt_config

Output

updated content container (XML) false on error

Description

Set link content in container and database. The content container will be returned and not saved.

7.3.7 setcomplink

Syntax

setcomplink (\$site, \$contentdata, \$contentfile, \$component_curr, \$component, \$condition, \$art, \$compuser, \$user)

Input parameters

\$site ... publication name
\$contentdata ... container (XML)
\$contentfile ... container name
\$component_curr ... component arrays (some are optional)
\$component ... article array or string [yes, no]
\$condition ... content user array or string
\$art ... user name
\$compuser
\$user

global input parameters

\$mgmt_config

Output

updated content container (XML) false on error

Description

Set component link content in container and database. The content container will be returned and not saved.

7.3.8 sethead

Syntax

sethead (\$site, \$contentdata, \$contentfile, \$headcontent, \$user, \$charset="")

Input parameters

\$site ... publication name \$contentdata ... container (XML) \$contentfile ... container name \$headcontent ... content array \$user ... user name \$charset ... chracter set of text content

global input parameters

\$mgmt_config

Output

updated content container (XML) false on error

Description

Only used for content in general head information of container.

7.3.9 setfilename

Syntax

setfilename (\$filedata, \$tagname, \$value)

Input parameters

\$filedata ... file content \$tagname ... hyperCMS tag name in page or component [content \$value ... template

Output

filedata/false on error

Description

Sets or creates the file name of the hyperCMS content file, template file, media file or file name pointer

7.3.10 setboxes

Syntax

setboxes (\$name_array, \$user)

Input parameters

\$name_array ... home box names as array or string \$user ... user name

global input parameters

\$mgmt_config

Output

true / false

7.4 Connect API Functions

7.4.1 ftp_userlogon

Syntax

ftp_userlogon (\$server, \$user, \$passwd, \$ssl=false)

Input parameters

\$server ... FTP servername or IP \$user ... user name \$passwd ... password \$ssl ... SSL [true, false] (optional)

global input parameters

\$mgmt_config

Output

true / false on error

Description

This function connects and performs logon to an FTP server

7.4.2 ftp_userlogout

Syntax

ftp_userlogout (\$conn_id)

Input parameters

\$conn_id ... FTP connection

global input parameters

\$mgmt_config

Output

true / false on error

Description

This function disconnects from an FTP server

7.4.3 ftp_getfile

Syntax

ftp_getfile (\$conn_id, \$remote_file, \$local_file, \$passive=true)

Input parameters

\$conn_id ... FTP connection
\$remote_file ... path to file on FTP server
\$local_file ... passive mode [true, false] (optional)
\$passive

global input parameters

\$mgmt_config

Output

true / false on error

Description

This function gets a file from the FTP server

7.4.4 ftp_putfile

Syntax

ftp_putfile (\$conn_id, \$local_file, \$remote_file, \$passive=true)

Input parameters

\$conn_id ... FTP connection \$local_file ... path to local file \$remote_file ... path to file on FTP server \$passive ... passive mode [true, false] (optional)

global input parameters

\$mgmt_config

Output

true / false on error

Description

This function puts a file to the FTP server

7.4.5 ftp_filelist

Syntax

ftp_filelist (\$conn_id, \$path=".", \$passive=true)

Input parameters

\$conn_id ... FTP connection
\$path ... path to remote directory (optional)
\$passive ... passive mode [true, false] (optional)

global input parameters

\$mgmt_config

Output

result array / false on error

Description

This function gets a file/directory listing of the FTP server

7.4.6 createsharelink_facebook

Syntax

createsharelink_facebook (\$site, \$url)

Input parameters

\$site ... URL to share \$url

global input parameters

\$mgmt_config

Output

Share URL / false on error

7.4.7 createsharelink_twitter

Syntax

createsharelink_twitter (\$site, \$url, \$text)

Input parameters

\$site ... URL to share \$url ... message to share \$text

global input parameters

\$mgmt_config

Output

Share URL / false on error

7.4.8 createsharelink_googleplus

Syntax

createsharelink_googleplus (\$site, \$url)

Input parameters

\$site ... URL to share

global input parameters

\$mgmt_config

Output

Share URL / false on error

7.4.9 createsharelink_linkedin

Syntax

createsharelink_linkedin (\$site, \$url, \$title, \$summary, \$source)

Input parameters

\$site ... URL to share \$url ... title \$title ... summary (optional) \$summary ... source (optional) \$source

global input parameters

\$mgmt_config

Output

Share URL / false on error

7.4.10 createsharelink_pinterest

Syntax

createsharelink_pinterest (\$site, \$image_url, \$title, \$description)

Input parameters

\$site ... image URL to share \$image_url ... title \$title ... description (optional) \$description

global input parameters

\$mgmt_config

Output

Share URL / false on error

7.5 Security API Functions

7.5.1 rootpermission

Syntax

rootpermission (\$site_name, \$site_admin, \$permission_str)

Input parameters

\$site_name ... publication name
\$site_admin ... publication admin
\$permission_str ... permission string from group

global input parameters

\$rootpermission

\$mgmt_config

Output

global permission array/false

Description

Deseralizes the permission string and and returns the root permission array

7.5.2 global permission

Syntax

globalpermission (\$site_name, \$permission_str)

Input parameters

\$site_name ... publication \$permission_str ... permission string from group

Output

global permission array/false

Description

Deseralizes the permission string and returns the global permission array

7.5.3 localpermission

Syntax

localpermission (\$site_name, \$permission_str)

Input parameters

\$site_name ... publication
\$permission_str ... permission string from group

Output

local permission array/false

Description

Deseralizes the permission string and returns the local permission array

7.5.4 accessgeneral

Syntax

accessgeneral (\$site, \$location, \$cat)

Input parameters

\$site ... publication\$location ... location (path to folder)\$cat ... object category ['page

global input parameters

\$mgmt_config
\$hiddenfolder
\$siteaccess

Output

true/false

Description

Checks general access to certain system folders, publications and returns true if access is granted

7.5.5 accesspermission

Syntax

accesspermission (\$site, \$location, \$cat)

Input parameters

\$site ... location (path to folder) \$location ... object category ['page \$cat ... comp']

global input parameters

\$pageaccess \$compaccess \$hiddenfolder \$hcms_linking \$mgmt_config

Output

group with access permissions as array / false on error

Description

Evaluates page and component access permissions and returns group(s)

7.5.6 setlocal permission

Syntax

setlocalpermission (\$site, \$group_array, \$cat)

Input parameters

\$site ... publication \$group_array ... group name array \$cat ... object category [page, comp]

global input parameters

\$localpermission

Output

local permission array / false on error

Description

Sets local permissions of a user group for a specific publication

7.5.7 checkpublicationpermission

Syntax

checkpublicationpermission (\$site, \$strict=true)

Input parameters

\$site ... publication name

\$strict ... strictly limited to siteaccess only without inheritance [true/false] (optional)

global input parameters

\$mgmt_config
\$siteaccess

Output

"direct" for direct access via group permission / "inherited" for access through inheritance / false

Description

Checks access to a publication based on the site access and inheritance settings

7.5.8 checkadminpermission

Syntax

checkadminpermission ()

Input parameters

global input parameters

\$adminpermission

Output

true/false

Description

Checks super admin permission

7.5.9 checkrootpermission

Syntax

checkrootpermission (\$name)

Input parameters

\$name ... permission name

global input parameters

\$rootpermission

Output

true/false

Description

Checks root permission

7.5.10 checkglobalpermission

Syntax

checkglobalpermission (\$site, \$name)

Input parameters

\$site ... publication name \$name ... permission name

global input parameters

\$globalpermission

Output

true/false

Description

Checks global permission for a publication

7.5.11 checklocalpermission

Syntax

checklocalpermission (\$site, \$group, \$name)

Input parameters

\$site ... publication name \$group ... user group name \$name ... permission name

global input parameters

\$\$localpermission

Output

true/false

Description

Checks local permissions of a user group for a specific publication

7.5.12 userlogin

Syntax

userlogin (\$user, \$passwd, \$hash="", \$objref="", \$objcode="", \$ignore_password=false, \$locking=true)

Input parameters

\$user ... username
\$passwd ... password
\$hash ... hash code of user
\$objref ... object reference for hcms linking (object ID)
\$objcode ... object code for hcms linking (crypted object ID)
\$ignore_password ... ignore passwordcheck needed for WebDAV or access link [true/false]
\$locking ... lock IP after 10 failed attempts to login [true/false]

global input parameters

\$mgmt_config
\$eventsystem
\$hcms_lang_codepage
\$hcms_lang
\$lang

Output

result array

Description

Login of user by sending user and password using the variables: \$sentuser, \$sentpasswd This procedure will register the user in the hypercms session and in the php session. The procedure will return true or false using the variable \$result.

7.5.13 registerinstance

Syntax

registerinstance (\$instance, \$load_config=true)

Input parameters

\$instance ... instance name

\$load_config ... load main config of instance [true/false] (optional)

global input parameters

\$mgmt_config

Output

true/false

7.5.14 createchecksum

Syntax

createchecksum (\$permissions="")

Input parameters

\$permissions ... array or empty

Output

MD5 checksum

Description

Creates the checksum of the user permissions

7.5.15 writesession

Syntax

writesession (\$user, \$passwd, \$checksum)

Input parameters

\$user ... user name \$passwd ... password \$checksum ... checksum

global input parameters

\$mgmt_config

Output

true / false on error

Description

Writes hyperCMS specific session data of a user

7.5.16 writesessiondata

Syntax

writesessiondata ()

Input parameters

global input parameters

\$mgmt_config

Output

true / false on error

Description

Serializes and writes all session data of a user

7.5.17 createsession

Syntax

createsession ()

Input parameters

global input parameters

\$mgmt_config

Output

true

Description

Checks if session data of a user is available. This function does access session variables directly!

7.5.18 killsession

Syntax

killsession (\$user="", \$destroy_php=true)

Input parameters

\$user ... user name for hyperCMS session (optional)
\$destroy_php ... destroy php session [true, false] (optional)

global input parameters

\$mgmt_config

Output

true

Description

Destroys session data of user

7.5.19 checkdiskkey

Syntax

checkdiskkey (\$users="", \$site="")

Input parameters

\$users ... user count (optional)
\$site ... publication names (use | as seperator) (optional)

global input parameters

\$mgmt_config

Output

true/false

Description

Checks the disc key of the installation

7.5.20 checkpassword

Syntax

checkpassword (\$password)

Input parameters

\$password ... password a string

global input parameters

\$mgmt_config
\$lang

Output

true if passed / error message as string

Description

This function checks the strength of a password and return the error messages or true

7.5.21 loguserip

Syntax

loguserip (\$client_ip, \$user="sys")

Input parameters

\$client_ip ... client IP address
\$user ... user logon name (optional)

global input parameters

\$mgmt_config

Output

true / false on error

7.5.22 checkuserip

Syntax

checkuserip (\$client_ip, \$user="", \$timeout="")

Input parameters

\$client_ip ... client IP address
\$user ... user logon name (optional)
\$timeout ... timeout in minutes (optional)

global input parameters

\$mgmt_config

Output

true if IP is not locked / false if IP is locked or on error

7.5.23 checkuserrequests

Syntax

checkuserrequests (\$user="sys")

Input parameters

\$user ... user name (optional)

global input parameters

\$mgmt_config

Output

true / false if a certain amount of reguests per minute is exceeded

Description

Provides security for Cross-Site Request Forgery

7.5.24 checkusersession

Syntax

checkusersession (\$user="sys", \$CSRF_detection=true)

Input parameters

\$user ... user name (optional)
\$CSRF_detection ... include CSRF detection [true, false]

global input parameters

\$mgmt_config

Output

true / html-output followed by termination requires config.inc.php

Description

Checks if session data of user is correct. This function does access session variables directly!

7.5.25 allowuserip

Syntax

allowuserip (\$site)

Input parameters

\$site ... publication name

global input parameters

\$mgmt_config

Output

true / false requires config.inc.php

Description

Verifies if the client IP is in the range of valid IPs and logs IP addresses with no access

7.5.26 valid_objectname

Syntax

valid_objectname (\$variable)

Input parameters

\$variable ... variable (string or array)

Output

true / false

Description

Checks if an object name includes forbidden characters in order to prevent directory browsing

7.5.27 valid_locationname

Syntax

valid_locationname (\$variable)

Input parameters

\$variable ... variable (string or array)

Output

true / false

Description

Checks if an location includes forbidden characters in order to prevent directory browsing

7.5.28 valid_publicationname

Syntax

valid_publicationname (\$variable)

Input parameters

\$variable ... variable (string or array)

Output

true / false

Description

Checks if a publication name includes forbidden characters in order to prevent directory browsing

7.5.29 html encode

Syntax

html_encode (\$expression, \$encoding="", \$js_protection=false)

Input parameters

\$expression ... variable as string or array

\$encoding ... conversion of all special characters based on given character set or to ASCII (optional)

\$js_protection ... remove characters to avoid JS injection [true, false] (optional)

Output

html encoded value as array or string / false on error

Description

This function encodes certain characters (&, <, >, ", ') into their HTML character entity equivalents to protect against XSS.

Converts a string into the html equivalents (also used for XSS protection).

Supports multibyte character sets like UTF-8 as well based on the ASCII value of the character.

7.5.30 html_decode

Syntax

html_decode (\$expression, \$encoding="")

Input parameters

\$expression ... epxression as string or array

\$encoding ... conversion of all special characters based on character set (optional)

Output

html decoded value as array or string / false on error

Description

This function decodes all characters which have been converted by html_encode

7.5.31 scriptcode_encode

Syntax

scriptcode_encode (\$content)

Input parameters

\$content ... content as string

global input parameters

\$mgmt_config

Output

escaped content as string / false on error

Description

This function escapes all script tags.

This function must be used to clean all user input in the CMS by removing all server side scripts tags.

7.5.32 scriptcode_extract

Syntax

scriptcode_extract (\$content, \$identifier_start="<?", \$identifier_end="?>")

Input parameters

\$content ... content as string \$identifier_start ... identifier of script begin \$identifier_end ... and end

Output

script code as array / false on error or if noting was found

Description

This function extracts the script code of a given content

7.5.33 scriptcode_clean_functions

Syntax

scriptcode_clean_functions (\$content, \$type=3, \$application="PHP")

Input parameters

\$content ... content as string
\$type ... cleaning level type from none = 0 to strong = 3 (no cleaning = 0
\$application ... basic set of disabled functions = 1

global input parameters

\$mgmt_config

Output

result array / false on error

Description

This function removes all dangerous PHP functions

7.5.34 sql_clean_functions

Syntax

sql_clean_functions (\$content)

Input parameters

\$content ... SQL statement as string

global input parameters

\$mgmt_config

Output

result array / false on error

Description

This function checks SQL statements for write operations

7.5.35 url_encode

Syntax

url_encode (\$variable)

Input parameters

\$variable ... variable as string or array

global input parameters

\$mgmt_config

Output

urlencoded value as array or string / false on error

Description

This function encodes all characters

7.5.36 url_decode

Syntax

url_decode (\$variable)

Input parameters

\$variable ... variable as string or array

global input parameters

\$mgmt_config

Output

urldecoded value as array or string / false on error

Description

This function decodes all characters which have been converted by url_encode or urlencode (PHP)

7.5.37 shellcmd_encode

Syntax

shellcmd_encode (\$variable)

Input parameters

\$variable ... variable as string or array

Output

encoded value as array or string / false on error

Description

This function encodes/escapes characters to secure the shell comand

7.5.38 hcms_crypt

Syntax

hcms_crypt (\$string, \$start=0, \$length=0)

Input parameters

\$string ... string to encode \$start ... start position \$length | length for string 6

\$length ... length for string extraction

global input parameters

\$mgmt_config

Output

encoded string / false on error

Description

Unidrectional encryption using crc32 and urlencode. Used to create tokens for simple view links in the system.

The tokens can be verified by calculating the hash of the media file name and comparing the hash values.

Don't use this function to secure any string or to for password hashing.

7.5.39 hcms_encrypt

Syntax

hcms_encrypt (\$string, \$key="", \$crypt_level="", \$encoding="url")

Input parameters

\$string ... string to encode
\$key ... key of length 16 or 24 or 32 (optional)
\$crypt_level ... crypt strength level [weak, standard, strong] (optional)
\$encoding ... encoding [base64, url, none] (optional)

global input parameters

\$mgmt_config

Output

encoded string / false on error

Description

Encryption of a string. Only strong encryption is binary-safe!

7.5.40 hcms_decrypt

Syntax

hcms_decrypt (\$string, \$key="", \$crypt_level="", \$encoding="url")

Input parameters

\$string ... hash-string to decode
\$key ... key of length 16 or 24 or 32 (optional)
\$crypt_level ... crypt strength level [weak, standard, strong] (optional)
\$encoding ... encoding [base64, url, none] (optional)

global input parameters

\$mgmt_config

Output

decoded string / false on error

Description

Decryption of a string. Only strong encryption is binary-safe!

7.5.41 createtimetoken

Syntax

createtimetoken (\$lifetime=0, \$secret=4)

Input parameters

\$lifetime ... token lifetime in seconds (optional) \$secret ... secret value (optional)

global input parameters

\$mgmt_config

Output

token / false on error

7.5.42 checktimetoken

Syntax

checktimetoken (\$token, \$secret=4)

Input parameters

\$token ... token

\$secret ... secret value (optional)

global input parameters

\$mgmt_config

Output

true / false

7.5.43 createtoken

Syntax

createtoken (\$user="sys", \$lifetime=0, \$secret=4)

Input parameters

\$user ... user name (optional)
\$lifetime ... token lifetime in seconds (optional)
\$secret ... secret value (optional)

global input parameters

\$mgmt_config

Output

token / false on error

7.5.44 checktoken

Syntax

checktoken (\$token, \$user="sys", \$secret=4)

Input parameters

\$token ... token

\$user ... user name (optional)
\$secret ... secret value (optional)

global input parameters

\$mgmt_config

Output

true / false

7.5.45 createuniquetoken

Syntax

createuniquetoken (\$length=16)

Input parameters

\$length ... token length (optional)

global input parameters

\$mgmt_config

Output

token as string / false

7.5.46 rand_secure

Syntax

rand_secure (\$min=1000, \$max=9999999999)

Input parameters

\$min ... min and max value as integer (optional) \$max

Output

secure random number / false

7.6 Media API Functions

7.6.1 indexcontent

Syntax

indexcontent (\$site, \$location, \$file, \$container="", \$container_content="", \$user)

Input parameters

\$site ... publication name

\$location ... path to multimedia file

\$file ... multimedia file name (file to be indexed)

\$container ... container name or ID (optional)

\$container_content ... container XML-content (optional)
\$user ... user name

global input parameters

\$mgmt_config
\$mgmt_parser
\$mgmt_imagepreview
\$mgmt_uncompress
\$hcms_ext
\$hcms_lang
\$lang

Output

true / false

Description

This function extracts the text content of multimedia objects and writes it the text to the container.

The given charset of the publication (not set by default), container or publication (not set by default) will be used.

The default character set of default.meta.tpl is UTF-8, so all content should be saved in UTF-8

7.6.2 unindexcontent

Syntax

unindexcontent (\$site, \$location, \$file, \$container, \$container_content, \$user)

Input parameters

\$site ... publication name \$location ... file location \$file ... file name \$container ... multimedia file to index \$container_content ... container name or ID \$user ... container XML-content

global input parameters

\$mgmt_config
\$mgmt_parser
\$hcms_lang
\$lang

Output

true / false

Description

This function removes media objects from the container

7.6.3 reindexcontent

Syntax

reindexcontent (\$site, \$container_id_array="")

Input parameters

\$site ... publication name
\$container_id_array ... container IDs as array (optional)

global input parameters

\$mgmt_config

Output

true / false

Description

Reindexes all media files of a publication and only for specific containers.

7.6.4 createthumbnail_indesign

Syntax

createthumbnail_indesign (\$site, \$location_source, \$location_dest, \$file)

Input parameters

\$site ... publication \$location_source ... path to source dir \$location_dest ... path to destination dir \$file ... file name

global input parameters

\$mgmt_config
\$user

Output

new file name / false on error (saves only thumbnail media file in destination location only jpeg format is supported as output)

Description

Creates a thumbnail by extracting the thumbnail from an indesign file and transferes the generated image via remoteclient.

For good results, InDesign Preferences must be set to save preview image and at extra large size.

7.6.5 createthumbnail_video

Syntax

createthumbnail_video (\$site, \$location_source, \$location_dest, \$file, \$frame, \$autorotate=false)

Input parameters

\$site ... publication \$location_source ... path to source dir \$location_dest ... path to destination dir \$file ... file name \$frame ... frame of video in the seconds or hh:mm:ss[.xxx] \$autorotate ... autorotate [true, false] (optional)

global input parameters

\$mgmt_config
\$mgmt_mediapreview
\$mgmt_mediaoptions
\$user

Output

new file name / false on error (saves only thumbnail media file in destination location only jpeg format is supported as output)

Description

Creates a thumbnail picture of a video frame

7.6.6 createmedia

Syntax

createmedia (\$site, \$location_source, \$location_dest, \$file, \$format="", \$type="thumbnail",
\$force_no_encrypt=false)

Input parameters

\$site ... publication

\$location_source ... path to source dir

\$location_dest ... path to destination dir

\$file ... file name

\$format ... format (file extension w/o dot) (optional)

\$type ... type of image/video/audio file [thumbnail(for thumbnails of images),

origithumb(thumbnail made from original video/audio), original(to overwrite original video/audio file), any other string present in \$mgmt_imageoptions/\$mgmt_mediaoptions] (optional)

\$force_no_encrypt ... force the file to be not encrypted even if the content of the publication must be encrypted [true, false] (optional)

global input parameters

\$mgmt_config

\$mgmt_imagepreview

\$mgmt mediapreview

\$mgmt_mediaoptions

\$mgmt_imageoptions

\$mgmt_maxsizepreview

\$mgmt_mediametadata

\$hcms_ext

\$user

Output

new file name / false on error

Description

Creates an new image from original or creates a thumbnail and transferes the generated image via remoteclient.

Saves original or thumbnail media file in destination location, for thumbnail only jpeg format is supported as output.

7.6.7 convertmedia

Syntax

convertmedia (\$site, \$location_source, \$location_dest, \$mediafile, \$format, \$media_config="", \$force_no_encrypt=false)

Input parameters

\$site ... publication name

\$location_source ... path to source dir

\$location_dest ... path to destination dir

\$mediafile ... file name

\$format ... target format (file extension w/o dot) of destination file

\$media_config ... media configuration to be used (optional)

\$force_no_encrypt ... force the file to be not encrypted even if the content of the publication must be encrypted [true, false] (optional)

global input parameters

\$mgmt_config

\$mgmt_imagepreview

\$mgmt_mediapreview

\$mgmt_mediaoptions

\$mgmt_imageoptions

\$mgmt_maxsizepreview

\$mgmt_mediametadata

\$hcms_ext

Output

new file name / false on error

Description

Converts and creates a new image/video/audio or document from original. This is a wrapper function for createmedia and createdocument.

7.6.8 convertimage

Syntax

convertimage (\$site, \$file_source, \$location_dest, \$format="jpg", \$colorspace="RGB", \$iccprofile="", \$width="", \$height="", \$slug=0, \$units="px", \$dpi=72, \$quality="")

Input parameters

\$site ... publication name

\$file_source ... path to source image file

\$location_dest ... path to destination dir

\$format ... format (file extension w/o dot) of destination file (optional)

\$colorspace ... colorspace of new image [CMY, CMYK, Gray, HCL, HCLp, HSB, HSI, HSL, HSV, HWB, Lab, LCHab, LCHuv, LMS, Log, Luv, OHTA, Rec601YCbCr, Rec709YCbCr, RGB, scRGB,

sRGB, Transparent, XYZ, YCbCr, YCC, YDbDr, YIQ, YPbPr, YUV] (optional)

\$iccprofile ... width in pixel/mm/inch (optional)

\$width ... height in pixel/mm/inch (optional)

\$height ... slug in pixel/mm/inch (optional)

\$slug ... units for width

\$units ... height and slug [px, mm, inch] (optional)

\$dpi ... dpi (optional)

\$quality ... image quality (1 to 100)

global input parameters

\$mgmt_config

\$mgmt_imagepreview

\$mgmt mediapreview

\$mgmt_mediaoptions

\$mgmt_imageoptions

\$mgmt_maxsizepreview

\$mgmt_mediametadata

\$hcms_ext

\$user

Output

new file name / false on error

Description

Converts and creates a new image from original. The new image keeps will be resized and cropped to fit width and height.

This is a wrapper function for createmedia.

7.6.9 rotateimage

Syntax

rotateimage (\$site, \$filepath, \$angle, \$imageformat)

Input parameters

\$site ... publication \$filepath ... path to source media file \$angle ... rotation angle

\$imageformat ... destination image format [jpg, png, gif]

global input parameters

\$mgmt_config

Output

new image file name / false on error

Description

Rotates an image (must be jpg, png or gif) using GD library. not used if ImageMagick is available

7.6.10 getimagecolors

Syntax

getimagecolors (\$site, \$file)

Input parameters

\$site ... publication \$file ... media file name

global input parameters

\$mgmt_config
\$user

Output

result array / false on error

Description

Uses the thumbnail image to calculate the mean color (red, green, blue), defines the colorkey (5 most commonly used colors) and the image type (landscape, portrait, square)

7.6.11 getimagecolorkey

Syntax

getimagecolorkey (\$image)

Input parameters

\$image ... image resource

global input parameters

\$mgmt_config

Output

color key of image / false on error

Description

Extracts the color key for an image that represents the 5 mostly used colors: K...black

W...white

E...grey

R...red

G...green

B...blue

C...cyan

M...magenta

Y...yellow

O...orange

P...pink

N...brown

7.6.12 hex2rgb

Syntax

hex2rgb (\$hex)

Input parameters

\$hex ... image color as hex-code

Output

RGB-color as array / false on error

7.6.13 rgb2hex

Syntax

rgb2hex (\$red, \$green, \$blue)

Input parameters

\$red ... image color in RGB \$green \$blue

Output

hex-color as string / false on error

7.6.14 readmediaplayer_config

Syntax

readmediaplayer_config (\$location, \$configfile)

Input parameters

\$location ... path to media config file \$configfile ... config file name

global input parameters

\$mgmt_config \$user

Output

config array / false on error

7.6.15 savemediaplayer_config

Syntax

savemediaplayer_config (\$location, \$configfile, \$mediafiles, \$width=320, \$height=240, \$rotation="", \$filesize="", \$duration="", \$videobitrate="", \$audiobitrate="",

\$audiofrequenzy="", \$audiochannels="", \$video_codec="", \$audio_codec="")

Input parameters

\$location ... path to media config file

\$configfile ... media config file name

\$mediafiles ... media file name array or string

\$width ... width in px (optional)

\$height ... height in px (optional)

\$rotation ... rotation in degree (optional)

\$filesize ... file size in kB (optional)

\$duration ... duration in hh: mmm:ss (optional)

\$videobitrate ... video bitrate in kb/s (optional)

\$audiobitrate ... audio bitrate in kb/s (optional)

\$audiofrequenzy ... audio frequenzy in Hz (optional)

\$audiochannels ... audio channels [mono

\$video_codec ... stereo] (optional)

\$audio_codec ... video codec name (optional)

global input parameters

\$mgmt_config

\$user

Output

true / false on error

7.6.16 createdocument

Syntax

createdocument (\$site, \$location_source, \$location_dest, \$file, \$format="",
\$force_no_encrypt=false)

Input parameters

\$site ... publication

\$location_source ... path to source location

\$location_dest ... path to destination location

\$file ... file name

\$format ... destination file format (extension w/o dot)

\$force_no_encrypt ... force the file to be not encrypted even if the content of the publication must be encrypted [true, false] (optional)

global input parameters

\$mgmt_config

\$mamt docpreview

\$mgmt_docoptions

\$mgmt_docconvert

\$mgmt_maxsizepreview

\$hcms_ext

\$hcms_lang

\$lang

\$user

Output

new file name / false on error

Description

Creates a new multimedia file of given format at source destination using UNOCONV and saves it as thumbnail file at the desitnation location

7.6.17 unzipfile

Syntax

unzipfile (\$site, \$zipfilepath, \$location, \$filename, \$cat="comp", \$user)

Input parameters

\$site ... publication \$zipfilepath ... path to source zip file \$location ... path to destination location \$filename ... category [page, comp] \$cat ... name of file for extraction \$user ... user

global input parameters

\$mgmt_config \$mgmt_uncompress \$mgmt_imagepreview \$mgmt_mediapreview \$mgmt_mediaoptions

Output

result array with all object paths / false

Description

Unpacks ZIP file and creates media files in destination location for components or unzips files directly for pages (not recommended due to securoty risks by uplaoding unsecured files).

7.6.18 clonefolder

Syntax

clonefolder (\$site, \$source, \$destination, \$user, \$activity="")

Input parameters

\$site ... publication name \$source ... source location \$destination ... destination location \$user ... user name

\$activity ... activity that need to be set for daily stats [download] (optional)

global input parameters

\$mgmt_config \$pageaccess \$compaccess \$hiddenfolder \$hcms_linking \$globalpermission \$setlocalpermission

Output

true/false

Description

Help function for function zipfiles that reads all multimedia files from their multimedia objects and copies them to the same folder structure using the object names.

7.6.19 zipfiles

Syntax

zipfiles (\$site, \$multiobject_array, \$destination="", \$zipfilename, \$user, \$activity="")

Input parameters

\$site ... publication

\$multiobject_array ... array with path to source zip files

\$destination ... destination location (if this is null then the \$location where the zip-file resists will be used)

\$zipfilename ... name of ZIP-file

\$user ... user name

\$activity ... activity that need to be set for daily stats [download] (optional)

global input parameters

\$mgmt_config

\$mgmt_compress

\$pageaccess

\$compaccess

\$hiddenfolder

\$hcms_linking

\$globalpermission

\$setlocalpermission

Output

true/false

Description

Compresses all media files and includes their folder structure in a ZIP file.

7.6.20 px2mm

Syntax

px2mm (\$pixel, \$dpi=72)

Input parameters

\$pixel ... pixel

\$dpi ... dpi (optional)

Output

pixel / false

Description

Convert mm to pixel

7.6.21 px2inch

Syntax

px2inch (\$pixel, \$dpi=72)

Input parameters

\$pixel ... pixel

\$dpi ... dpi (optional)

Output

inch / false

Description

Convert pixel to inches

7.6.22 inch2px

Syntax

inch2px (\$inch, \$dpi=72)

Input parameters

\$inch ... pixel \$dpi ... dpi (optional)

Output

pixel / false

Description

Convert inches to pixel

7.6.23 mediasize2frame

Syntax

mediasize2frame (\$mediawidth, \$mediaheight, \$framewidth="", \$frameheight="", \$keepmaxsize=true)

Input parameters

\$mediawidth ... media width \$mediaheight ... media height

\$framewidth ... frame width (optional) \$frameheight ... frame height (optional)

\$keepmaxsize ... keep maximum media size based on original dimensions of media without stretching [true, false] (optional)

Output

width and height as array / false

Description

Calculates the width and height of a media to fit into a given frame size.

7.6.24 vtt2array

Syntax

vtt2array (\$vtt)

Input parameters

\$vtt ... VTT string

Output

array / false

Description

Converts VTT string to array

7.7 Metadata API Functions

7.7.1 importmetadata

Syntax

importmetadata (\$site, \$location, \$file, \$user, \$type="", \$delimiter=";", \$enclosure="", \$charset="utf-8")

Input parameters

\$site ... publication name
\$location ... location
\$file ... path to CSV file
\$user ... user name
\$type ... type array or string [u, f, l, c, d, k] (optional)
\$delimiter ... delimiter (optional)
\$enclosure ... enclosure (optional)
\$charset ... character set (optional)

global input parameters

\$mgmt_config
\$eventsystem

Output

true / false

Description

Imports metadata form CSV file for various assets linked by name or conatiner ID. Empty rows or rows without a delimiter will be ignored.

In order to identify an asset the file name as "Name" or the container ID as "containerID" must be provided in the first row before the content rows.

7.7.2 createtaxonomy

Syntax

createtaxonomy (\$recreate=false)

Input parameters

\$recreate ... recreate taxonomy file [true, false] (optional)

global input parameters

\$mgmt_config

Output

true / false

Description

Generates a PHP array from a taxonomy defintion file and saves the PHP file in data/include/publication-name.taxonomy.inc.php

7.7.3 splitkeywords

Syntax

splitkeywords (\$keywords, \$charset="UTF-8")

Input parameters

\$keywords ... comma seperated keywords string \$charset ... character set (optional)

Output

keywords as array / false on error

Description

Generates a keyword list from a text by splitting and transforming the comma seperated string.

7.7.4 copymetadata

Syntax

copymetadata (\$file_source, \$file_dest)

Input parameters

\$file_source ... path to source file \$file_dest ... path to destination file

global input parameters

\$mgmt_config
\$mgmt_mediametadata
\$user

Output

true / false

Description

Copies all meta data from source to destination file using EXIFTOOL

7.7.5 extractmetadata

Syntax

extractmetadata (\$file)

Input parameters

\$file ... path to image file

global input parameters

\$user \$mgmt_config \$mgmt_mediametadata

Output

result array / false on error

Description

Extracts all meta data from a file using EXIFTOOL

7.7.6 xmlobject2array

Syntax

xmlobject2array (\$obj, \$namespace="")

Input parameters

\$obj ... XML as object\$namespace ... namespace as array (optional)

Output

result array / false

Description

Converts an xmlobject to an array, provided by xaviered at gmail dot com

7.7.7 id3_getdata

Syntax

id3_getdata (\$file)

Input parameters

\$file ... path to audio file

global input parameters

\$mgmt_config
\$hcms_ext

Output

result array / false on error

Description

Requires getID3 library since EXIFTOOL cannot write ID3 tags so far

7.7.8 id3_writefile

Syntax

id3_writefile (\$file, \$id3, \$keep_data=true, \$movetempfile=true)

Input parameters

\$file ... abs. path to audio file \$id3 ... ID3 tag array

\$keep_data ... keep existing ID3 data of file [true, false] (optional)

\$movetempfile ... move tempoarary file from unecrypted to encrypted [true, false] (optional)

global input parameters

\$user \$mgmt_config \$mgmt_mediametadata \$hcms_ext

Output

true / false on error

Description

Writes ID3 tags into audio file for supported file types and keeps the existing ID3 tags

7.7.9 id3_create

Syntax

id3_create (\$site, \$text)

Input parameters

\$site ... publication name

\$text ... text array (from content container)

global input parameters

\$mgmt_config

Output

ID3 tag array / false on error

Description

Defines ID3 tag array based on the media mapping of a publication.

7.7.10 xmp_getdata

Syntax

xmp_getdata (\$file)

Input parameters

\$file ... path to image file

global input parameters

\$user
\$mgmt_config
\$hcms_ext

Output

result array / false on error

7.7.11 xmp_writefile

Syntax

xmp_writefile (\$file, \$xmp, \$keep_data=true, \$movetempfile=true)

Input parameters

\$file ... abs. path to image file \$xmp ... XMP tag array \$keep_data ... keep existing XMP data of file [true, false] (optional) \$movetempfile ... move tempoarary file from unecrypted to encrypted [true, false] (optional)

global input parameters

\$user \$mgmt_config \$mgmt_mediametadata \$hcms_ext

Output

true / false on error

Description

Writes XMP tags into image file for supported file types and keeps the existing XMP tags

7.7.12 xmp_create

Syntax

xmp_create (\$site, \$text)

Input parameters

\$site ... publication name
\$text ... text array (from content container)

global input parameters

\$mgmt_config

Output

XMP tag array / false on error

Description

Defines XMP tag array based on the media mapping of a publication

7.7.13 geo2decimal

Syntax

geo2decimal (\$deg, \$min, \$sec, \$hemi)

Input parameters

\$deg ... geo location in degree

\$min ... minutes \$sec ... seconds

\$hemi ... hemisphere [N, O, S, W]

Output

decimal result / false

7.7.14 exif_getdata

Syntax

exif_getdata (\$file)

Input parameters

\$file ... path to image file

global input parameters

\$user
\$mgmt_config
\$hcms_ext

Output

result array / false

7.7.15 iptc_getdata

Syntax

iptc_getdata (\$file)

Input parameters

\$file ... path to image file

global input parameters

\$user
\$mgmt_config
\$hcms_ext

Output

result array / false

7.7.16 iptc_getcharset

Syntax

iptc_getcharset (\$tag)

Input parameters

\$tag ... iptc tag that holds character set information

Output

charset as string / false on error

Description

Copied from MediaWiki!

Warning, this function does not (and is not intended to) detect all iso 2022 escape codes. In practise, the code for utf-8 is the only code that seems to have wide use. It does detect that code.

According to iim standard, charset is defined by the tag 1:90.

in which there are iso 2022 escape sequences to specify the character set.

the iim standard seems to encourage that all necessary escape sequences are in the 1:90 tag, but says it doesn't have to be.

This is in need of more testing probably. This is definitely not complete.

however reading the docs of some other iptc software, it appears that most iptc software only recognizes utf-8. If 1:90 tag is not present content is

usually ascii or iso-8859-1 (and sometimes utf-8), but no guarantee.

This also won't work if there are more than one escape sequence in the 1:90 tag or if something is put in the G2, or G3 charsets, etc. It will only reliably recognize utf-8. This is just going through the charsets mentioned in appendix C of the iim standard.

7.7.17 iptc_maketag

Syntax

iptc_maketag (\$record=2, \$tag, \$value)

Input parameters

\$record ... type of tag (e.g. 2) \$tag ... code of tag (e.g. 025) \$value ... value of tag

Output

binary IPTC tag / false on error

Description

Convert the IPTC tag to binary code

7.7.18 iptc_writefile

Syntax

iptc_writefile (\$file, \$iptc, \$keep_data=true, \$movetempfile=true)

Input parameters

\$file ... abs. path to image file

\$iptc ... IPTC tag array

\$keep_data ... keep existing IPTC data of file [true, false] (optional)

\$movetempfile ... move tempoarary file from unecrypted to encrypted [true, false] (optional)

global input parameters

\$user

\$mgmt_config

\$mgmt_mediametadata

Output

true / false on error

Description

Writes IPTC tags into image file for supported file types and keeps the existing IPTC tags

7.7.19 iptc_create

Syntax

iptc_create (\$site, \$text)

Input parameters

\$site ... publication name

\$text ... text array (from content container)

global input parameters

\$mgmt_config

Output

IPTC tag array / false on error

Description

Defines IPTC tag array based on the medai mapping of a publication

7.7.20 createmapping

Syntax

createmapping (\$site, \$mapping)

Input parameters

\$site ... publication name \$mapping ... mapping definition

global input parameters

\$mgmt_config

Output

true / false on error

Description

Prepares the PHP mapping array from the provided mapping definition and saves media mapping file

7.7.21 getmapping

Syntax

getmapping (\$site)

Input parameters

\$site ... publication name

global input parameters

\$mgmt_config

Output

mapping code for display / false

Description

Load the mapping file of the provided publication.

7.7.22 setmetadata

Syntax

```
setmetadata ($site, $location="", $object="", $mediafile="", $mapping="", $containerdata="", $user, $savecontainer=true)
```

Input parameters

\$site ... publication name

\$location ... location path (optional)

\$object ... object name (optional)

\$mediafile ... media file name (optional)

\$mapping ... mapping array [meta data tag name -> text-id] (optional)

\$containerdata ... container content as XML string (optional)

\$user ... user name

\$savecontainer ... save content container [true, false] (optional)

global input parameters

\$eventsystem \$mgmt_config \$hcms_ext

Output

container content as XML string / false

Description

Saves meta data of a multimedia file using a provided mapping in the proper fields of the content container.

If no mapping is given a default mapping will be used.

7.8 Link API Functions

7.8.1 link_db_restore

Syntax

link_db_restore (\$site="")

Input parameters

\$site ... publication name (optional)

global input parameters

\$mgmt_config

Output

true / false on error

Description

This function restores a given or all link management index files

7.8.2 link_db_load

Syntax

link_db_load (\$site, \$user)

Input parameters

\$site ... publication name

\$user ... user name

global input parameters

\$mgmt_config

Output

link database [2 dim. array] or true / false on error

Description

This function loads and locks the link management database

each record of the link management database has the following design: xml-content container: | absolute path to 1-n objects: | 1-m links used by 1-n objects important: the link management database has to saved or closed after loading it.

7.8.3 link_db_read

Syntax

link_db_read (\$site)

Input parameters

\$site ... site

global input parameters

\$mgmt_config

Output

link database [2 dim. array] or true / false on error

Description

This function loads the link management database for reading without locking

7.8.4 link_db_close

Syntax

link_db_close (\$site, \$user)

Input parameters

\$site ... publication name \$user ... user name

global input parameters

\$mgmt_config

Output

true/false

Description

closes and unlocks the link management database.

7.8.5 link_db_save

Syntax

link_db_save (\$site, \$link_db, \$user)

Input parameters

\$site ... link database array
\$link_db ... site
\$user ... user

global input parameters

\$mgmt_config

Output

true/false on error

Description

This function saves und unlocks the link management database

7.8.6 link db update

Syntax

link_db_update (\$site, \$link_db, \$attribute, \$contentfile, \$cat, \$link_curr, \$link_new, \$option)

Input parameters

\$site ... publication name \$link_db ... link database [2 dim. array] \$attribute ... attribute ['object' \$contentfile ... 'link'] \$cat ... content container [optional] [string] \$link_curr ... link category [optional] ['comp' \$link_new ... 'page'] \$option ... current link [optional

global input parameters

\$mgmt_config

Output

link database [2 dim. array] or true / false on error

Description

This function inserts, updates and removes objects and their links from the link management database (add or update a link)

depending on which link is left empty:

link_curr = "": add new link (just one link matching given category!)

link_new = "": delete current link in use (just one linkm matching given category!)

link_curr & link_new are not empty: update current link with the new one

7.8.7 link_db_insert

Syntax

link_db_insert (\$site, \$link_db, \$contentfile, \$cat, \$object)

Input parameters

\$site ... publication name
\$link_db ... link database [2 dim. array]
\$contentfile ... content container
\$cat ... link category ['comp
\$object ... page']

global input parameters

\$mgmt_config

Output

link database [2 dim. array] or true / false

Description

This function inserts a new record in the link management database optionally the created object can be also inserted

7.8.8 link_db_delete

Syntax

link_db_delete (\$site, \$link_db, \$contentfile)

Input parameters

\$site ... link database [2 dim. array] \$link_db ... content container \$contentfile

global input parameters

\$mgmt_config

Output

link database [2 dim. array] or true / false on error

Description

This function deletes a record in the link management database

7.8.9 link_db_getobject

Syntax

link_db_getobject (\$multiobject)

Input parameters

\$multiobject ... link database attribut (references to objects seperated by |)

global input parameters

\$mgmt_config

Output

objects [array] / false on error

Description

This function splits the object string into an array of objects.

7.8.10 link_update

Syntax

link_update (\$site, \$container, \$link_old, \$link_new)

Input parameters

\$site ... publication name
\$container ... container name
\$link_old ... old link (converted)
\$link_new ... new link (converted)

global input parameters

\$user

\$mgmt_config

Output

true/false

Description

This function updates the link of the published and working content container and link file

7.8.11 getlinkedobject

Syntax

getlinkedobject (\$site, \$location, \$page, \$cat)

Input parameters

\$site ... publication

\$location ... location

\$page ... object (name and extension)

\$cat ... category [page

global input parameters

\$mgmt_config

Output

objects which link to the given object [array] or true / false

Description

This function gets all objects which link to the given object.

works with pages (page links) and components (component links) if link management is enabled.

7.8.12 getconnectedobject

Syntax

getconnectedobject (\$container, \$type="work")

Input parameters

\$container ... container name

\$type ... container type [work, published, version] (optional)

global input parameters

\$mgmt_config

\$user

Output

connected objects[array]

Description

This function gets all objects which use the same content container and are therefore connected.

7.8.13 extractlinks

Syntax

extractlinks (\$textcontent, \$identifier)

Input parameters

\$textcontent ... text content as string

\$identifier ... link identifiert ("href" for hyperreferences

global input parameters

\$mgmt_config

Output

object links [array] / false on error

Description

This function extracts all links based on it's identifier from a text and returns an array of all links

7.9 Plugin API Functions

7.9.1 plugin_getdefaultconf

Syntax

plugin_getdefaultconf ()

Input parameters

Output

default value as array

7.9.2 plugin_readmenu

Syntax

plugin_readmenu (\$xml, \$pluginFolder)

Input parameters

\$xml ... plugin xml as string\$pluginFolder ... plugin directory

global input parameters

\$mgmt_config

Output

menu point array used by navigator

Description

Reads Menupoints and menugroups from the xml data.

Be carefull with nesting, function getcontent is used and you can't nest groups inside of groups as a subpoint!

Input parameter \$pluginFolder contains the directory of the plugin.

Returns an Array containing every group and menupoint with its configuration.

7.9.3 plugin_parse

Syntax

plugin_parse (\$oldData=array())

Input parameters

\$oldData ... mgmt_plugin as array (optional)

global input parameters

\$mgmt_config

Output

mgmt_plugin as array

Description

Reads the plugin configurations from the file system.

Checks the folder defined in mgmt_config and searched for plugins and their configurations files.

It either takes needed values from the configuration, from the \$oldData or defaultConfiguration.

7.9.4 plugin_generatedefinition

Syntax

plugin_generatedefinition (\$arrayName, \$array)

Input parameters

\$arrayName ... name of array holding the plugin definitions \$array ... configuration array containing the values and keys

global input parameters

\$mgmt_config

Output

plugin array / false on error

Description

Generates the Array definition used in php for \$array with the name of \$arrayName. Run recursively through the array and supports boolean, numeric and string types for the key and value.

7.9.5 plugin_saveconfig

Syntax

plugin_saveconfig (\$configuration)

Input parameters

\$configuration ... configuration as array

global input parameters

\$mgmt_config

Output

true / false on error

Description

Saves the plugin configuration \$configuration into the configuration file.

The configuration file is located in the data/config directory and is named plugin.conf.php.

7.9.6 plugin_generatelink

Syntax

plugin_generatelink (\$plugin, \$page, \$control=false, \$additionalGetParameters=false)

Input parameters

\$plugin ... plugin name

\$page ... plugin page (relative reference to the plugins main page)

\$control ... control (relative reference to the plugins control page)

\$additionalGetParameters ... additional GET parameters

global input parameters

\$mgmt_config

Output

plugin link

Description

Generates a link to be used when linking to other pages inside of a plugin

7.10 User Interface API Functions

7.10.1 windowwidth

Syntax

windowwidth (\$type="object")

Input parameters

\$type ... type [object] (optional)

global input parameters

\$mgmt_config

Output

window width in pixels

Description

Returns the width of the object window when editing/opening an object

7.10.2 windowheight

Syntax

windowheight (\$type="object")

Input parameters

\$type ... type [object] (optional)

global input parameters

\$mgmt_config

Output

window height in pixels

Description

Returns the height of the object window when editing/opening an object

7.10.3 toggleview

Syntax

toggleview (\$view)

Input parameters

\$view ... view [detail, small, medium, large]

global input parameters

\$mgmt_config

Output

true / false

Description

Sets explorer objectlist view parameter

7.10.4 togglesidebar

Syntax

togglesidebar (\$view)

Input parameters

\$view ... view [true, false]

global input parameters

\$mgmt_config

Output

true / false

Description

Enables or disables the sidebar

7.10.5 setfilter

Syntax

setfilter (\$filter_set)

Input parameters

\$filter_set ... set of filters as array with keys [comp, image, document, video, audio] and value [0, 1]

global input parameters

\$mgmt_config

Output

true / false

Description

Set filter settings for object view in session

7.10.6 objectfilter

Syntax

objectfilter (\$file)

Input parameters

\$file ... file name

global input parameters

\$mgmt_config
\$hcms_ext

Output

true / false

Description

If an object name is passing the filter-test. One or more filters need to be set in the session "hcms_objectfilter".

7.10.7 showshorttext

Syntax

showshorttext (\$text, \$length=0, \$linebreak=false, \$charset="UTF-8")

Input parameters

\$text ... text as string

\$length ... max. length of text (minus length starting from the end) (optional)

\$linebreak ... line break instead of cut [true, false] only if length is positive (optional) \$charset ... character set for encoding (optional)

Output

shortened text if possible or orignal text

Description

Reduce the length of a string and add "..." at the end

7.10.8 showtopbar

Syntax

showtopbar (\$show, \$lang="en", \$close_link="", \$close_target="", \$individual_button="", \$id="bar")

Input parameters

\$show ... message \$lang ... language code (optional) \$close_link ... close button link (optional) \$close_target ... link target (optional) \$individual_button ... individual button (optional) \$id ... ID of div-layer (optional)

global input parameters

\$mgmt_config
\$hcms_charset
\$hcms_lang

Output

top bar box / false on error

Description

Returns the standard top bar with or without close button

7.10.9 showtopmenubar

Syntax

showtopmenubar (\$show, \$menu_array, \$lang="en", \$close_link="", \$close_target="", \$id="bar")

Input parameters

\$show ... message \$menu_array ... menu as array [key=name \$lang ... value=properties/events] \$close_link ... language code (optional) \$close_target ... close button link (optional) \$id ... link target (optional)

global input parameters

\$mgmt_config
\$hcms_charset
\$hcms_lang

Output

top bar box / false on error

Description

Returns the menu top bar with or without close button

7.10.10 showmessage

Syntax

showmessage (\$show, \$width="580px", \$height="70px", \$lang="en", \$style="", \$id="hcms_messageLayer")

Input parameters

\$show ... message \$width ... width in pixel (optional) \$height ... height in pixel (optional) \$lang ... language code (optional) \$style ... additional style definitions of div-layer (optional) \$id ... ID of div-layer (optional)

global input parameters

\$mgmt_config
\$hcms_charset
\$hcms_lang

Output

message box / false on error

Description

Returns the standard message box with close button

7.10.11 showinfopage

Syntax

showinfopage (\$show, \$lang="en", \$onload="")

Input parameters

\$show ... message \$lang ... language code (optional) \$onload ... on load JS events (optional)

global input parameters

\$mgmt_config
\$hcms_charset
\$hcms_lang_codepage
\$hcms_lang

Output

message on html info page / false on error

Description

Returns a full html info page

7.10.12 showinfobox

Syntax

showinfobox (\$show, \$lang="en", \$style="", \$id="hcms_infoboxLayer")

Input parameters

\$show ... message \$lang ... language code (optional) \$style ... additional style definitions of div-layer (optional) \$id ... ID of div-layer (optional)

global input parameters

\$mgmt_config \$hcms_charset \$hcms_lang_codepage \$hcms_lang

Output

message in div layer / false on error

Description

Returns the infobox as long as it has not been closed. Saves the close event in localstorage of browser.

7.10.13 showsharelinks

Syntax

showsharelinks (\$link, \$mediafile, \$lang="en", \$style="", \$id="hcms_shareLayer")

Input parameters

\$link ... link to share \$mediafile ... media file name \$lang ... language code (optional) \$style ... additional style definitions of div-layer (optional) \$id ... ID of div-layer (optional)

global input parameters

\$mgmt_config \$hcms_charset \$hcms_lang_codepage \$hcms_lang

Output

message in div layer / false on error

Description

Returns the presenation of share links of social media platforms

7.10.14 showmetadata

Syntax

showmetadata (\$data, \$lang="en", \$class_headline="hcmsRowData2")

Input parameters

\$data ... meta data as array \$lang ... hierarchy level \$class_headline ... CSS-class with background-color for headlines (optional)

global input parameters

\$mgmt_config
\$hcms_charset
\$hcms_lang_codepage
\$hcms_lang

Output

result as HTML unordered list / false on error

7.10.15 showobject

Syntax

showobject (\$site, \$location, \$page, \$cat="", \$name="")

Input parameters

\$site ... publication name \$location ... location \$page ... object name \$cat __category [page co

\$cat ... category [page, comp] (optional)

\$name ... object name (optional)

global input parameters

\$mgmt_config \$hcms_charset \$hcms_lang \$lang

Output

html presentation / false

7.10.16 showmedia

Syntax

showmedia (\$mediafile, \$medianame, \$viewtype, \$id="", \$width="", \$height="", \$class="hcmsImageItem")

Input parameters

\$mediafile ... mediafile (publication/filename)
\$medianame ... name of mediafile for display
\$viewtype ... view type [template, media_only, preview, preview_download,
preview_no_rendering]
\$id ... ID of the HTML media tag
\$width ... width in px (optional)
\$height ... height in px (optional)
\$class ... CSS class (optional)

global input parameters

\$site

\$mgmt_config

\$mgmt_mediapreview

\$mgmt_mediaoptions

\$mgmt_imagepreview

\$mgmt_docpreview

\$mgmt_docconvert

\$hcms_charset

\$hcms_lang_codepage

\$hcms_lang

\$lang

\$site

\$location

\$cat

\$page

\$user

⊅u26i

\$pageaccess

\$compaccess

\$hiddenfolder

\$hcms_linking

\$setlocalpermission \$mgmt_imageoptions \$is_mobile \$is_iphone

Output

html presentation of any media asset / false

Description

This function requires site, location and cat to be set as global variable in order to validate the access permission of the user

7.10.17 showcompexplorer

Syntax

```
showcompexplorer ($site, $dir, $location_esc="", $page="", $compcat="multi", $search_expression="", $search_format="", $mediatype="", $lang="en", $callback="", $scalingfactor="1")
```

Input parameters

\$site ... publication name

\$dir ... current explorer location

\$location_esc ... object location (optional)

\$page ... object name (optional)

\$compcat ... component category [single, multi, media] (optional)

\$search_expression ... search expression (optional)

\$search_format ... search format [object, document, image, video, audio] (optional)

\$mediatype ... media-type [audio, video, text, flash, image, compressed, binary] (optional)

\$lang ... callback of CKEditor (optional)

\$callback ... saclingfactor for images (optional)

\$scalingfactor

global input parameters

\$user

\$mgmt_config

\$siteaccess

\$pageaccess

\$compaccess

\$rootpermission

\$globalpermission

\$localpermission

\$hiddenfolder

\$html5file

\$temp_complocation

\$hcms_charset

\$hcms_lang

Output

explorer with search / false on error

Description

Creates component explorer including the search form

7.10.18 showeditor

Syntax

```
showeditor ($site, $hypertagname, $id, $contentbot="", $sizewidth=600, $sizeheight=300, $toolbar="Default", $lang="en", $dpi=72)
```

Input parameters

\$site ... publication name
\$hypertagname ... hypertag name
\$id ... hypertag id
\$contentbot ... content
\$sizewidth ... width
\$sizeheight ... height of the editor
\$toolbar ... toolbar set
\$lang ... language
\$dpi ... dpi for scaling images

global input parameters

\$mgmt_config
\$publ_config

Output

rich text editor code / false on error

Description

Returns the rich text editor code

7.10.19 showinlineeditor_head

Syntax

showinlineeditor_head (\$lang)

Input parameters

\$lang ... language

global input parameters

\$mgmt_config
\$hcms_charset
\$hcms_lang

Output

rich text editor code for html head section / false on error

Description

Returns the rich text editor code (JS, CSS) for include into the html head section

7.10.20 showinlinedatepicker_head

Syntax

showinlinedatepicker_head ()

Input parameters

global input parameters

\$mgmt_config

Output

date picker code for html head section / false on error

Description

Returns the date picker code (JS, CSS) for include into the html head section

7.10.21 showinlineeditor

Syntax

showinlineeditor (\$site, \$hypertag, \$id, \$contentbot="", \$sizewidth=600, \$sizeheight=300, \$toolbar="Default", \$lang="en", \$contenttype="", \$cat="", \$location_esc="", \$page="", \$contentfile="", \$db_connect=0, \$token="")

Input parameters

\$site ... publication name
\$hypertag ... hypertag
\$id ... hypertag id
\$contentbot ... content
\$sizewidth ... width
\$sizeheight ... height of the editor
\$toolbar ... toolbar set
\$lang ... language
\$contenttype ... content-type
\$cat ... category[page, comp]
\$location_esc ... converted location
\$page ... object name
\$contentfile ... container name
\$db_connect ... DB-connect file name
\$token ... security token

global input parameters

\$mgmt_config \$publ_config \$hcms_charset \$hcms_lang

Output

rich text editor code / false on error

Description

shows the rich text inline editor

7.10.22 showvideoplayer

Syntax

showvideoplayer (\$site, \$video_array, \$width=320, \$height=240, \$logo_url="", \$id="", \$title="", \$autoplay=true, \$fullscreen=true, \$loop=false, \$muted=false, \$controls=true, \$iframe=false, \$force_reload=false)

Input parameters

\$site ... videoArray (Array) containing the different html sources

\$video_array ... width (Integer) Width of the video in pixel

\$width ... height (Integer) Height of the video in pixel

\$height ... logo_url (String) Link to the logo which is displayed before you click on play (If the value is null the default logo will be used)

\$logo_url ... id (String) The ID of the video (will be generated when empty)

\$id ... title (String) The title for this video

\$title ... autoplay (Boolean) Should the video be played on load (true)

\$autoplay ... default is false

\$fullscreen ... enableFullScreen (Boolean) Is it possible to view the video in fullScreen (true)

\$loop ... play loop (optional) [true, false]

\$muted ... muted/no sound (optional) [true, false]

\$controls ... player controls (optional) [true, false]

\$iframe ... use video in iframe (optional) [true, false]

\$force_reload ... reload video sources to prevent the browser cache to show the same video even if it has been changed [true, false] (optional)

global input parameters

\$mgmt_config

Output

HTML code of the video player / false on error

Description

Generates a html segment for the video player code

7.10.23 showvideoplayer_head

Syntax

showvideoplayer_head (\$secureHref=true, \$fullscreen=true)

Input parameters

\$secureHref ... secure hyperreferences by adding 'hypercms_'
\$fullscreen ... is it possible to view the video in fullScreen [true, false]

global input parameters

\$mgmt_config

Output

head for video player / false on error

7.10.24 showaudioplayer

Syntax

showaudioplayer (\$site, \$audioArray, \$width=320, \$height=320, \$logo_url="", \$id="", \$autoplay=false, \$loop=false, \$controls=true, \$force_reload=false)

Input parameters

\$site ... publication name
\$audioArray ... audio files as array (Array)
\$width ... ID of the tag (optional)
\$height ... autoplay (optional) [true, false]
\$logo_url ... play loop (optional) [true, false]
\$id ... player controls (optional) [true, false]
\$autoplay
\$loop
\$controls
\$force_reload

global input parameters

\$mgmt_config

Output

code of the HTML5 player / false

Description

Generates the html segment for the video player code

7.10.25 showaudioplayer_head

Syntax

showaudioplayer_head (\$secureHref=true)

Input parameters

\$secureHref ... secure hyperreferences by adding 'hypercms_'

global input parameters

\$mgmt_config

Output

head for audio player

7.10.26 debug_getbacktracestring

Syntax

debug_getbacktracestring (\$valueSeparator, \$rowSeparator, \$ignoreFunctions=array())

Input parameters

\$valueSeparator ... separator for arguments\$rowSeparator ... separator for a Row on screen/file\$ignoreFunctions ... functionnames to be ignored

Output

debug message

Description

Returns the current backtrace as a good readable string. Ignores debug and debug_getbacktracestring.

7.10.27 showAPIdocs

Syntax

showAPIdocs (\$file, \$return="html")

Input parameters

\$file ... path to API file \$return ... return result as HTML or array [html, array] (optional)

global input parameters

= array()

Output

HTML output of documentation / false on error

Description

Generates the documentation of an API file

7.10.28 readnavigation

Syntax

readnavigation (\$site, \$docroot, \$object, \$view="publish", \$user="sys")

Input parameters

\$site ... publication name
\$docroot ... location
\$object ... object name
\$view ... view [see view parameters of function buildview]
\$user ... user name

global input parameters

```
$mgmt_config
$navi_config
```

Output

navigation item array / false

Description

Reads the content from the container and collects information about a single navigation item

7.10.29 createnavigation

Syntax

```
createnavigation ($site, $docroot, $urlroot, $view="publish", $currentobject="",
$recursive=true)
```

Input parameters

```
$site ... publication name
$docroot ... document root for navigation
$urlroot ... URL root for navigation
$view ... view [see view parameters of function buildview]
$currentobject ... path to current object (optional)
$recursive ... recursive [true, false] (optional)
```

global input parameters

\$mgmt_config
\$navi_config

Output

navigation array / false

Description

```
Generates an associative array (item => nav-item, sub => array with sub-items).
Example
$navi config = array();
document root definitions
$navi_config['root_path'] = "%abs_page%/";
$navi_config['root_url'] = "%url_page%/";
HTML / CSS class defintions
$navi_config['attr_ul_top'] = "class=\"nav navbar-nav\"";
$navi_config['attr_ul_dropdown'] = "class=\"dropdown-menu\"";
$navi_config['attr_li_active'] = "class=\"active\"";
$navi_config['attr_li_dropdown'] = "class=\"dropdown\"";
$navi config['attr href dropdown'] = "class=\"dropdown-toggle\" data-
toggle=\"dropdown\"";
$navi_config['tag_li'] = "<a href=\"%link%\"</pre>
%attr_href%>%title%</a>%sub%\n";
$navi_config['tag_ul'] = "%list%\n";
Language definitions
Session variable name that holds the language setting
$navi_config['lang_session'] = "langcode";
2nd key = langcode & value = text_id of textnode
$navi_config['lang_text_id']['DE'] = "Titel_DE";
$navi_config['lang_text_id']['EN'] = "Titel_EN";
PermaLink definitions
2nd key = langcode & value = text_id of textnode
$navi_config['permalink_text_id']['DE'] = "PermaLink_DE";
$navi_config['permalink_text_id']['EN'] = "PermaLink_EN";
Hide navigation item (any value or empty) and use sort order (number or empty)
```

```
$navi_config['hide_text_id'] = "NavigationHide";
$navi_config['sort_text_id'] = "NavigationSortOrder";
Use only index file of directory as navigation item, e.g. index.html or index.php (Keep empty
if all objects of a folder should be included)
$navi_config['index_file'] = "";
$navigation = createnavigation ("%publication%", $navi_config['root_path'],
$navi_config['root_url'], "%view%", "%abs_location%/%object%");
echo shownavigation ($navigation);
```

7.10.30 shownavigation

Syntax

shownavigation (\$navigation, \$level=1)

Input parameters

\$navigation ... navigation array (created by function readnavigation) \$level ... level as integer (optional)

global input parameters

\$mamt config \$navi_config

Output

navigation HTML presentation / false

Description

```
display navigation as HTML code.
The following example configures the navigation:
$navi_config = array();
document root definitions:
$navi_config['root_path'] = "%abs_page%/";
$navi_config['root_url'] = "%url_page%/";
HTML / CSS class defintions (names between percentage signs are placeholders):
$navi config['attr ul top'] = "class=\"nav navbar-nav\"";
$navi_config['attr_ul_dropdown'] = "class=\"dropdown-menu\"";
$navi_config['attr_li_active'] = "class=\"active\"";
$navi_config['attr_li_dropdown'] = "class=\"dropdown\"";
$navi_config['attr_href_dropdown'] = "class=\"dropdown-toggle\" data-
toggle=\"dropdown\"";
$navi_config['tag_li'] = "<a href=\"%link%\"</pre>
%attr_href%>%title%</a>%sub%\n";
$navi_config['tag_ul'] = "%list%\n";
language definitions
Session variable name that holds the language setting
$navi_config['lang_session'] = "langcode";
note: key = langcode & value = text_id of textnode
$navi_config['lang_text_id']['DE'] = "Titel_DE";
$navi_config['lang_text_id']['EN'] = "Titel_EN";
PermaLink defintions
note: key = langcode & value = text_id of textnode
$navi_config['permalink_text_id']['DE'] = "PermaLink_DE";
$navi_config['permalink_text_id']['EN'] = "PermaLink_EN";
Navigation hide and sort order defintions
$navi_config['hide_text_id'] = "NavigationHide";
$navi_config['sort_text_id'] = "NavigationSortOrder";
Use the first object of a folder for the main navigation item and display all following objects as
sub navigation items [true, false]
$navi_config['use_1st_folderitem'] = false;
```

7.10.31 showselect

Syntax

showselect (\$value_array, \$only_text=false, \$selected_value="", \$id="", \$attributes="")

Input parameters

\$value_array ... values array (array-key = value
\$only_text ... array-value = text)
\$selected_value ... use values of array as option value and text [true, false] (optional)
\$id ... selected value (optional)
\$attributes ... attributes of select tags like name or id or events (optional)

Output

HTML select box presentation / false

7.10.32 showtranslator

Syntax

showtranslator (\$site, \$id, \$type, \$charset="UTF-8", \$lang="en", \$style="")

Input parameters

\$site ... publication name \$id ... editor/text-tag ID \$type ... unformatted or formatted texttag-type [u, f] \$charset ... character set (optional) \$lang ... language code (optional) \$style ... style of div tag (optional)

global input parameters

\$mgmt_config \$hcms_lang

Output

HTML translator box presentation / false

7.10.33 showmapping

Syntax

showmapping (\$site, \$lang="en")

Input parameters

\$site ... publication name \$lang ... language code (optional)

global input parameters

\$mgmt_config \$hcms_charset \$hcms_lang

Output

table with form fields for display / false

Description

Present the mapping form of the provided publication.

7.11 Template Engine API Functions

7.11.1 checklanguage

Syntax

checklanguage (\$language_array, \$language_value)

Input parameters

\$language_array ... language array with all valid values \$language_value ... language value of attribute in hyperCMS tag

Output

true if language array holds the given language value / false if not found

7.11.2 checkgroupaccess

Syntax

checkgroupaccess (\$groupaccess, \$ownergroup)

Input parameters

\$groupaccess ... group access string from hyperCMS group-tag attribute \$ownergroup ... owner groups as array

Output

true if current ownergroup has access or invalid input / false if not

7.11.3 transformlink

Syntax

transformlink (\$viewstore)

Input parameters

\$viewstore ... view of object

global input parameters

\$site \$location_esc \$page \$ctrlreload \$mgmt_config

Output

view with transformed links for easyedit mode

7.11.4 followlink

Syntax

followlink (\$site, \$follow)

Input parameters

\$site ... publication name \$follow ... link to follow

global input parameters

\$mgmt_config

Output

prepared input (location plus page) for easyedit mode (buildview) / false on error

7.11.5 errorhandler

Syntax

errorhandler (\$source_code, \$return_code, \$error_identifier)

Input parameters

\$source_code ... source code \$return_code ... return code \$error_identifier ... error identifier

Output

error message and view of the code with line identifiers

7.11.6 viewinclusions

Syntax

viewinclusions (\$site, \$viewstore, \$hypertag, \$view, \$application, \$charset="UTF-8")

Input parameters

\$site ... view of object

\$viewstore ... hypertag to create view of inlcuded objects

\$hypertag ... view parameter

\$view ... application

\$application ... character set used (optional) view-parameter explanation: \$view = "template or any other word" -> the standard text (in table) will be included for the view \$view = "preview" -> preview of the content of the included file \$view = "publish" -> view the content of the included file as ist is (for publishing)
\$charset

global input parameters

\$user \$mgmt_config \$location \$hcms_lang \$lang

Output

view on the content including the content of included objects

7.11.7 buildview

Syntax

buildview (\$site, \$location, \$page, \$user, \$buildview="template", \$ctrlreload="no", \$template="", \$container="", \$force_cat="", \$execute_code=true)

Input parameters

\$site ... publication name \$location ... location \$page ... object

\$user ... user

\$buildview ... view parameter (optional)

\$ctrlreload ... reload workplace control frame and add html & body tags if missing [yes, no] (optional)

\$template ... template name (optional)

\$container ... container name (optional)

\$force_cat ... force category to use different location path [page, comp] (optional) \$execute_code ... execute_code [true/false] (optional)

global input parameters

\$container_collection

\$eventsystem

\$db_connect

\$mgmt_config

\$siteaccess

\$adminpermission

\$setlocalpermission

\$token

\$is mobile

\$is_iphone

\$viewportwidth

\$mgmt_lang_shortcut_default

\$hcms_charset

\$hcms_lang_name

\$hcms_lang_shortcut

\$hcms_lang_codepage

\$hcms_lang_date

\$hcms_lang

\$lang

Output

result array with view of the content / false on error

Description

buildview parameter may have the following values:

\$buildview = "formedit": use form for content editing

\$buildview = "formmeta": use form for content viewing only for meta informations (tag-type must be meta)

\$buildview = "formlock": use form for content viewing

\$buildview = "cmsview": view of page based on template, includes hyperCMS specific code (buttons)

\$buildview = "inlineview": view of page based on template, includes hyperCMS specific code (buttons) and inline text editing

\$buildview = "publish": view of page for publishing based on template without CMS specific code (buttons)

\$buildview = "preview": view of page based on template for preview (inactive hyperlinks) without CMS specific code (buttons)

\$buildview = "template": view of template based on template for preview (inactive hyperlinks) without CMS specific code (buttons)

7.11.8 buildsearchform

Syntax

buildsearchform (\$site="", \$template="", \$report="", \$ownergroup="", \$css_display="inline-block")

Input parameters

\$site ... publication name (optional for report)

\$template ... template name (optional)

\$report ... or report name (optional)

\$ownergroup ... group access as array (optional)

\$css_display ... CSS display value for label tag (optional)

global input parameters

\$user
\$mgmt_config
\$mgmt_lang_shortcut_default
\$hcms_charset
\$hcms_lang_name
\$hcms_lang_shortcut
\$hcms_lang_codepage
\$hcms_lang_date
\$hcms_lang

Output

form view / false on error

7.11.9 buildbarchart

Syntax

buildbarchart (\$paper_name, \$paper_width=600, \$paper_height=300, \$paper_top=10, \$paper_left=40, \$x_axis, \$y1_axis, \$y2_axis="", \$y3_axis="", \$paper_style="", \$bar1_style="", \$bar2_style="", \$show_value=false)

Input parameters

\$paper_name ... name/id of paper
\$paper_width ... width of paper in pixel
\$paper_height ... height of paper in pixel
\$paper_top ... top space in pixel
\$paper_left ... left space in pixel
\$x_axis ... x-axis values as array
\$y1_axis ... y1-axis values as array
\$y2_axis ... y2-axis values as array (optional)
\$y3_axis ... y3-axis values as array (optional)
\$paper_style ... paper CSS style
\$bar1_style ... 1st bar chart CSS style
\$bar2_style ... 2nd bar chart CSS style
\$bar3_style ... 3rd bar chart CSS style
\$show_value ... show y-value in bar [true, false]

global input parameters

\$lang \$mgmt_config

Output

bar chart view / false on error

7.12 XML API Functions

7.12.1 valid_tagname

Syntax

valid_tagname (\$tagname)

Input parameters

\$tagname ... tag name

Output

true / false on error

Description

Verifies a tag name

7.12.2 setxmlparameter

Syntax

setxmlparameter (\$xmldata, \$parameter, \$value)

Input parameters

\$xmldata ... XML content container \$parameter ... paramater name \$value ... paramater value

Output

XML content container / false on error

Description

Set parameter values in XML declaration (e.g. encoding): encoding="UTF-8"

7.12.3 getcontent

Syntax

getcontent (\$xmldata, \$starttagname)

Input parameters

\$xmldata ... XML content container \$starttagname ... tag name

Output

result array with the content of the requested XML node (tag) / false on error

Description

<tagname>content</tagname>

Extracts the content between the given \$starttagname xml-tags.

Only this function will decode special characters (&, <, >) in the content and removes CDATA. Function getcontent will only decode values if they are non-xml and non_html. so content inside child nodes including tags won't be decoded.

Wild card character "*" can be used at the end of \$starttagname.

7.12.4 geticontent

Syntax

geticontent (\$xmldata, \$starttagname)

Input parameters

\$xmldata ... XML content container \$starttagname ... tag name

Output

result array with the content of the requested XML node (tag) / false on error

Description

CASE-Insensitive version (XML parser are however always case-sensitive!) <tagname>content</tagname>

Extracts the content between the given \$starttagname xml-tags.

Only this function will decode special characters (&, <, >) in the content and removes CDATA. getcontent will only decode values if they are non-xml and non_html. so content inside child

nodes including tags won't be decoded.
Wild card character "*" can be used at the end of \$starttagname

7.12.5 getxmlcontent

Syntax

getxmlcontent (\$xmldata, \$starttagname)

Input parameters

\$xmldata ... XML content container \$starttagname ... tag name

Output

result array with the content of the requested XML node (tag) / false on error

Description

<tagname>content</tagname>

Extracts the content together with the \$starttagname xml tags.

This function will NOT decode special characters like function getcontent! Wild card character "*" can be used at the end of \$starttagname.

7.12.6 getxmlicontent

Syntax

getxmlicontent (\$xmldata, \$starttagname)

Input parameters

\$xmldata ... XML content container \$starttagname ... tag name

Output

result array with the content of the requested XML node (tag) / false on error

Description

CASE-Insensitive version (XML parser are always case-sensitive!) <tagname>content</tagname>

Extracts the content together with the \$starttagname xml tags.

This function will NOT decode special characters like function getcontent! Wild card character "*" can be used at the end of \$starttagname.

7.12.7 selectcontent

Syntax

selectcontent (\$xmldata, \$starttagname, \$startcondtag, \$condvalue)

Input parameters

\$xmldata ... XML content container

\$starttagname ... tag name of requested XML node

\$startcondtag ... tag holding the conditional value inside the given starttagname

\$condvalue ... conditional value

Output

result array with the content of the requested XML node (tag) / false on error

Description

<tagname>

.

<condtag>condvalue</condtag>

.

</tagname>

Extracts the content between the given \$starttagname xml tags where the child xml tag \$startcondtag value is equal with the target value \$condvalue.

Wild card character "*" can be used at the end of \$starttagname.

Wild card character "*" can be used at begin and end of \$condvalue.

Be Aware: \$startcondtag must be a child of \$starttagname!

7.12.8 selecticontent

Syntax

selecticontent (\$xmldata, \$starttagname, \$startcondtag, \$condvalue)

Input parameters

\$xmldata ... XML content container

\$starttagname ... tag name of requested XML node

\$startcondtag ... tag holding the conditional value inside the given starttagname

\$condvalue ... conditional value

Output

result array with the content of the requested XML node (tag) / false on error

Description

CASE-Insensitive version (XML parser are always case-sensitive!)

<tagname>

.

<condtag>condvalue</condtag>

.

</tagname>

Extracts the content between the given \$starttagname xml tags where the child xml tag \$startcondtag value is equal with the target value \$condvalue.

Wild card character "*" can be used at the end of \$starttagname.

Wild card character "*" can be used at begin and end of \$condvalue.

Be Aware: \$startcondtag must be a child of \$starttagname!

7.12.9 selectxmlcontent

Syntax

selectxmlcontent (\$xmldata, \$starttagname, \$startcondtag, \$condvalue)

Input parameters

\$xmldata ... XML content container

\$starttagname ... tag name of requested XML node

\$startcondtag ... tag holding the conditional value inside the given starttagname

\$condvalue ... conditional value

Output

result array with the content of the requested XML node (tag) / false on error

Description

<tagname>

.

<condtag>condvalue</condtag>

.

</tagname>

Extracts the content between the given \$starttagname xml tags where the child xml tag \$startcondtag value is equal with the target value \$condvalue

Wild card character "*" can be used at begin and end of \$condvalue.

Be Aware: \$startcondtag must be a child of \$starttagname!

7.12.10 selectxmlicontent

Syntax

selectxmlicontent (\$xmldata, \$starttagname, \$startcondtag, \$condvalue)

Input parameters

\$xmldata ... XML content container

\$starttagname ... tag name of requested XML node

\$startcondtag ... tag holding the conditional value inside the given starttagname

\$condvalue ... conditional value

Output

result array with the content of the requested XML node (tag) / false on error

Description

CASE-Insensitive version (XML parser are always case-sensitive!)

<tagname>

.

<condtag>condvalue</condtag>

.

</tagname>

Extracts the content between the given \$starttagname xml tags where the child xml tag \$startcondtag value is equal with the target value \$condvalue.

Wild card character "*" can be used at begin and end of \$condvalue.

Be Aware: \$startcondtag must be a child of \$starttagname!

7.12.11 deletecontent

Syntax

deletecontent (\$xmldata, \$starttagname, \$startcondtag, \$condvalue)

Input parameters

\$xmldata ... XML content container

\$starttagname ... tag name of requested XML node

\$startcondtag ... tag holding the conditional value inside the given starttagname

\$condvalue ... conditional value

Output

XML content container / false on error

Description

<tagname>

<condtag>condvalue</condtag>

</tagname>

Deletes the whole xml content including <tagname>.

Wild card character "*" can be used at begin and end of \$condvalue.

7.12.12 deleteicontent

Svntax

deleteicontent (\$xmldata, \$starttagname, \$startcondtag, \$condvalue)

Input parameters

\$xmldata ... XML content container

\$starttagname ... tag name of requested XML node

\$startcondtag ... tag holding the conditional value inside the given starttagname \$condvalue ... conditional value

Output

XML content container / false on error

Description

CASE-Insensitive version (XML parser are always case-sensitive!)

<tagname>

<condtag>condvalue</condtag>

</tagname>

Deletes the whole xml content including <tagname>.

Wild card character "*" can be used at begin and end of \$condvalue.

7.12.13 setcontent

Syntax

setcontent (\$xmldata, \$startparenttagname, \$starttagname, \$contentnew, \$startcondtag="", \$condvalue="")

Input parameters

\$xmldata ... XML content container

\$startparenttagname ... parent tag name

\$starttagname ... tag name of XML node for the new content

\$contentnew ... new XML node to be inserted

\$startcondtag ... tag holding the conditional value inside the given starttagname

\$condvalue ... conditional value

Output

XML content container / false on error

Description

<parenttagname>

<condtag>condvalue</condtag>

<tagname>contentnew</tagname>

</parenttagname>

\$xmldata = data string to be parsed

\$startparenttagname = name of the tag that is a parent node of starttagname (necessary if condition has been set!)

\$starttagname = name of the tag (child node)

\$contentnew = the content that will be inserted between the child tags \$starttagname

\$startcondtag = child xml tag where condition will be set

\$condvalue = value of the condition

Wild card character "*" can be used at begin and end of \$condvalue.

7.12.14 seticontent

Syntax

seticontent (\$xmldata, \$startparenttagname, \$starttagname, \$contentnew, \$startcondtag, \$condvalue)

Input parameters

\$xmldata ... XML content container

\$startparenttagname ... parent tag name

\$starttagname ... tag name of XML node for the new content

\$contentnew ... new XML node to be inserted

\$startcondtag ... tag holding the conditional value inside the given starttagname

\$condvalue ... conditional value

Output

XML content container / false on error

Description

CASE-Insensitive version (XML parser are always case-sensitive!)

- <parenttagname>
- <condtag>condvalue</condtag>
- <tagname>contentnew</tagname>
- </parenttagname>

\$xmldata = data string to be parsed

\$startparenttagname = name of the tag that is the parent node of starttagname (necessary if condition has been set!)

\$starttagname = name of the tag (child node)

\$contentnew = the content that will be inserted between the child tags \$starttagname

\$startcondtag = child xml tag where condition will be set

\$condvalue = value of the condition

Wild card character "*" can be used at begin and end of \$condvalue.

7.12.15 setcontent_fast

Syntax

setcontent_fast (\$xmldata, \$startparenttagname, \$starttagname, \$contentnew, \$startcondtag="", \$condvalue="")

Input parameters

\$xmldata ... XML content container

\$startparenttagname ... parent tag name

\$starttagname ... tag name of XML node for the new content

\$contentnew ... new XML node to be inserted

\$startcondtag ... tag holding the conditional value inside the given starttagname

\$condvalue ... conditional value

Output

XML content container / false on error

Description

function designed for link management, extremely fast but with limitations (only CASE-Sensitive!)

- <parenttagname>
- <condtag>condvalue</condtag>
- <tagname>contentnew</tagname>
- </parenttagname>

\$xmldata = data string to be parsed

\$startparenttagname = name of the tag that is the parent node of starttagname (necessary if condition has been set!)

\$starttagname = name of the tag (child node)

\$contentnew = the content that will be inserted between the child tags \$starttagname

\$startcondtag = child xml tag where condition will be set

\$condvalue = value of the condition

Wild card character "*" can be used at begin and end of \$condvalue.

7.12.16 updatecontent

Syntax

updatecontent (\$xmldata, \$xmlnode, \$xmlnodenew)

Input parameters

\$xmIdata ... XML content container \$xmInode ... XML node to be replaced \$xmInodenew ... new XML node

Output

XML content container / false on error

Description

Updates a given xml string \$xmlnode in \$xmldata with the content \$xmlnodenew. This method provides a faster way to update xml nodes when the node was selected before.

7.12.17 insertcontent

Syntax

insertcontent (\$xmldata, \$insertxmldata, \$starttagname)

Input parameters

\$xmIdata ... XML content container \$insertxmIdata ... XML node to be inserted in starttagname \$starttagname ... tag name of the parent XML node

Output

XML content container / false on error

Description

<tagname> <- list start
.....
insertxmldata <- insertxmldata
</tagname> <- list end
.....</pre>

Inserts \$insertxmldata string at the end of all child between the parent \$tagname .

7.12.18 inserticontent

Syntax

inserticontent (\$xmldata, \$insertxmldata, \$starttagname)

Input parameters

\$xmldata ... XML content container \$insertxmldata ... XML node to be inserted in starttagname \$starttagname ... tag name of the parent XML node

Output

XML content container / false on error

Description

Inserts \$insertxmldata string at the end of all child between the parent \$tagname.

7.12.19 addcontent

Syntax

addcontent (\$xmldata, \$sub_xmldata, \$startgrandtagname, \$startcondtag, \$condvalue, \$startparenttagname, \$starttagname, \$contentnew)

Input parameters

\$xmldata ... XML content container \$sub_xmldata ... xml node to be inserted \$startgrandtagname ... grandparent tag name \$startcondtag ... tag holding the conditional value inside the given starttagname \$condvalue ... conditional value \$startparenttagname ... parent tag name \$starttagname ... tag name of XML node for the new content \$contentnew ... new XML node to be inserted

Output

XML content container / false on error

Description

```
<grandtagname>
<condtag>condvalue</condtag>
<parenttagname> <- list start</pre>
......
<tagname>contentnew</tagname> } <- sub_xmldata
</parenttagname> <- list end
</grandtagname>
$xmldata = data string to be parsed
$sub xmldata = xml node to be inserted
$startgrandtagname (optional) = name of the grand xml tag of parent xml tag where (article)
$startcondtag (optional) = xml tag inside the parent xml tags where condition will be set
$condvalue (optional) = value of the condition
$startparenttagname (optional) = name of the parent xml tag where the xml subschema
should be added (list)
$starttagname (optional) = name of the tag (child)
$contentnew (optional) = the content that will be inserted between the child tags
```

7.12.20 addicontent

Syntax

addicontent (\$xmldata, \$sub_xmldata, \$startgrandtagname, \$startcondtag, \$condvalue, \$startparenttagname, \$starttagname, \$contentnew)

Input parameters

\$xmldata ... XML content container \$sub_xmldata ... xml node to be inserted \$startgrandtagname ... grandparent tag name \$startcondtag ... tag holding the conditional value inside the given starttagname \$condvalue ... conditional value \$startparenttagname ... parent tag name \$starttagname ... tag name of XML node for the new content \$contentnew ... new XML node to be inserted

Output

XML content container / false on error

Description

```
CASE-Insensitive version (XML parser are always case-sensitive!)
<grandtagname>
<condtag>condvalue</condtag>
<parenttagname> <- list start</pre>
......
......}
<tagname>contentnew</tagname> } <- sub_xmldata
.....}
</parenttagname> <- list end
</grandtagname>
$xmldata = data string to be parsed
$sub_xmldata = xml subschema to be inserted
$startgrandtagname (optional) = name of the grand xml tag of parent xml tag where (article)
$startcondtag (optional) = xml tag inside the parent xml tags where condition will be set
$condvalue (optional) = value of the condition
$startparenttagname (optional) = name of the parent xml tag where the xml subschema
should be added (list)
$starttagname (optional) = name of the tag (child)
$contentnew (optional) = the content that will be inserted between the child tags
```

7.13 Report API Functions

7.13.1 createreport

Syntax

createreport (\$report_name)

Input parameters

\$report_name ... report name

global input parameters

\$user \$eventsystem \$mgmt_config \$hcms_lang \$lang

Output

result array

Description

This function creates a new report

7.13.2 editreport

Syntax

editreport (\$report_name, \$config)

Input parameters

\$report_name ... report name
\$config ... report configuration as array

global input parameters

\$user \$eventsystem \$mgmt_config \$hcms_lang \$lang

Output

result array

Description

This function saves the configuration of a report.

7.13.3 loadreport

Syntax

loadreport (\$report_name)

Input parameters

\$report_name ... report name

global input parameters

\$user \$eventsystem \$mgmt_config \$hcms_lang \$lang

Output

result array / false

Description

This function loads the report configuration file and provides the data as array.

7.13.4 deletereport

Syntax

deletereport (\$report_name)

Input parameters

\$report_name ... report name

global input parameters

\$user \$eventsystem \$mgmt_config \$hcms_lang \$lang

Output

result array

Description

This function deletes a report

7.13.5 analyzeSQLselect

Syntax

analyzeSQLselect (\$sql)

Input parameters

\$sql ... SQL statement as string

Output

result array

Description

This function analyzes an SQL Select statement and return its parts in an array

7.13.6 showpiechart

Syntax

showpiechart (\$title="", \$x_title, \$x_value, \$y_title, \$y_value, \$data_array, \$width="100%", \$height="100%", \$id="hcmsPieChart", \$suffix="", \$loadAPI=true)

Input parameters

\$title ... chart title

\$x_title ... chart x-axis title

\$x_value ... chart x-axis array key name holding the values

\$y_title ... chart y-axis title

\$y_value ... chart y-axis array key name holding the values

\$data_array ... assoz. data array

\$width ... chart width in pixels (optional)

\$height ... chart height in pixels (optional)

\$id ... ID of chart (optional)

\$suffix ... function name suffix (optional)

\$loadAPI ... load Google Chart API [true, false] (optional)

global input parameters

\$mgmt_config
\$hcms_lang

\$lang

Output

Google Chart code / false

Description

This function creates the Google Chart Code for a pie chart. The Google Chart API need to be loaded first!

7.13.7 showcolumnchart

Syntax

showcolumnchart (\$title="", \$x_title, \$x_value, \$y1_title, \$y1_value, \$y2_title="", \$y2_value="", \$y3_title="", \$y3_value="", \$data_array, \$width="100%", \$height="100%", \$id="hcmsColumnChart", \$suffix="", \$loadAPI=true)

Input parameters

\$title ... chart title

\$x_title ... chart x-axis title

\$x_value ... chart x-axis array key name holding the values

\$y1_title ... chart y1-axis title

\$y1_value ... chart y1-axis array key name holding the values

\$y2_title ... chart y2-axis title

\$y2_value ... chart y2-axis array key name holding the values

\$y3_title ... chart y3-axis title

\$y3_value ... chart y3-axis array key name holding the values \$data_array ... assoz. data array \$width ... chart width in pixels (optional) \$height ... chart height in pixels (optional) \$id ... ID of chart (optional) \$suffix ... function name suffix (optional) \$loadAPI

global input parameters

\$mgmt_config
\$hcms_lang
\$lang

Output

Google Chart code / false

Description

This function creates the Google Chart Code for a column chart. The Google Chart API need to be loaded first!

7.13.8 showtimelinechart

Syntax

showtimelinechart (\$title="", \$y_title, \$y_value, \$x1_title, \$x1_value, \$x2_title="", \$x2_value="", \$data_array, \$width="100%", \$height="100%", \$id="hcmsTimelineChart", \$suffix="", \$loadAPI=true)

Input parameters

\$title ... chart title
\$y_title ... chart y-axis title
\$y_value ... chart y-axis array key name holding the values
\$x1_title ... chart x1-axis title
\$x1_value ... chart x1-axis array key name holding the values
\$x2_title ... chart x2-axis title
\$x2_value ... chart x2-axis array key name holding the values
\$x2_value ... chart x2-axis array key name holding the values
\$data_array ... assoz. data array
\$width ... chart width in pixels (optional)
\$height ... chart height in pixels (optional)
\$id ... ID of chart (optional)
\$suffix ... function name suffix (optional)
\$loadAPI ... load Google Chart API [true, false] (optional)

global input parameters

\$mgmt_config \$hcms_lang \$lang

Output

Google Chart code / false

Description

This function creates the Google Chart Code for a timeline chart. The Google Chart API need to be loaded first!

7.13.9 showgeolocationchart

Syntax

showgeolocationchart (\$title="", \$marker_value, \$lat_value, \$lng_value, \$link_value,

\$data_array, \$width="100%", \$height="100%", \$id="hcmsGeolocationChart", \$suffix="", \$loadAPI=true)

Input parameters

\$title ... chart title

\$marker_value ... array key name holding the value for the marker title
\$lat_value ... array key name holding the value for the latitude of the marker
\$lng_value ... array key name holding the value for the longitude of the marker
\$link_value ... array key name holding the value for the marker link
\$data_array ... assoz. data array
\$width ... chart width in pixels (optional)
\$height ... chart height in pixels (optional)
\$id ... ID of chart (optional)
\$suffix ... function name suffix (optional)
\$loadAPI ... load Google Maps API [true, false] (optional)

global input parameters

\$mgmt_config \$hcms_lang \$lang

Output

Google Chart code / false

Description

This function creates the Google Maps Code for a geolocation chart. The Google Maps API need to be loaded first!

7.14 Project API Functions

7.14.1 createproject

Syntax

createproject (\$subproject_id=0, \$object_id="", \$user, \$projectname, \$description="")

Input parameters

\$subproject_id ... ID of main project (only if the project is a subproject)
\$object_id ... object ID or path to object (optional)
\$user ... user name of sub/project owner
\$projectname ... project name
\$description ... project description (optional)

global input parameters

\$mgmt_config
\$hcms_lang_codepage
\$hcms_lang
\$lang

Output

result array

Description

This function creates a new project.

7.14.2 editproject

Syntax

editproject (\$project_id, \$subproject_id=0, \$object_id="", \$user="", \$projectname="",

\$description="")

Input parameters

\$project_id ... project ID
\$subproject_id ... ID of main project (only if project is a subproject)
\$object_id ... object ID or path to object (optional)
\$user ... user name of sub/project owner (optional)
\$projectname ... project name (optional)
\$description ... project description (optional)

global input parameters

\$mgmt_config
\$hcms_lang_codepage
\$hcms_lang
\$lang

Output

result array

Description

This function saves data of an exitsing project

7.14.3 deleteproject

Syntax

deleteproject (\$project_id)

Input parameters

\$project_id ... project ID or array of project IDs to be deleted

global input parameters

\$mgmt_config
\$hcms_lang
\$lang

Output

result array

Description

This function removes projects.

7.14.4 getprojectstructure

Syntax

getprojectstructure (\$project_id)

Input parameters

\$project_id ... project ID

global input parameters

\$mgmt_config

Output

assoziative result array / false

Description

This function creates an assoziatve array presenting the project structure (project -> subprojects -> tasks)

7.14.5 showprojectrecord

Syntax

showprojectrecord (\$project_record, \$masterproject_id)

Input parameters

\$project_record ... project record as array (project/subproject/task)
\$masterproject_id ... ID of master project

global input parameters

\$mgmt_config
\$hcms_lang
\$lang
\$siteaccess
\$user_array

Output

html presentation of the record / false

Description

This function creates a table row for the projects presentation

7.14.6 showganttchart

Syntax

showganttchart (\$title="", \$task_title, \$task_value, \$resource_title="", \$resource_value="", \$start_title, \$start_value, \$end_title="", \$end_value="", \$status_title="", \$status_value="", \$data_array, \$width="100%", \$height="100%", \$id="hcmsTimelineChart", \$suffix="", \$loadAPI=true)

Input parameters

\$title ... chart title \$task_title ... task title \$task_value ... task array key name holding the values \$resource_title ... resouce title (optional) \$resource_value ... resource array key name holding the values (optional) \$start_title ... task start date title \$start_value ... task start date array key name holding the values \$end_title ... task finish date title \$end_value ... task finish date array key name holding the values \$status_title ... work status title (optional) \$status_value ... work status array key name holding the values (optional) \$data_array ... assoz. data array \$width ... chart width in pixels (optional) \$height ... chart height in pixels (optional) \$id ... ID of chart (optional) \$suffix ... function name suffix (optional) \$loadAPI ... load Google Chart API [true, false] (optional)

global input parameters

\$mgmt_config \$hcms_lang \$lang

Output

Google Chart code / false

Description

This function creates the Google Chart Code for a Gantt chart. The Google Chart API need to be loaded first!

7.15 Task API Functions

7.15.1 createtask

Syntax

createtask (\$site="", \$from_user, \$from_email="", \$to_user, \$to_email="", \$startdate="", \$finishdate="", \$category="", \$object_id="", \$taskname, \$message="", \$sendmail=true, \$priority="low", \$project_id=0, \$planned="")

Input parameters

```
$site ... publication name (optional)
$from_user ... from_user name
$from_email ... from_email [email-address] (optional)
$to_user ... to_user name
$to_email ... to_email [email-address] (optional)
$startdate ... start date (optional)
$finishdate ... finish date (optional)
$category ... category [link, user, workflow] (optional)
$object_id ... object ID or object path
$taskname ... task name
$message ... message (optional)
$sendmail ... sendmail [true/false]
$priority ... priority [high, medium, low] (optional)
$project_id ... project/subproject ID if the task should be assigned to a project (optional)
$planned ... planned effort in taskunit (optional)
```

global input parameters

\$mgmt_config
\$hcms_lang_codepage
\$hcms_lang
\$lang

Output

result array

Description

Creates a new user task and send optional e-mail to user.

Since verion 5.8.4 the data will be stored in RDBMS instead of XML files.

7.15.2 edittask

Syntax

```
edittask ($task_id, $object_id="", $to_user="", $startdate="", $finishdate="", $taskname="", $message="", $sendmail=true, $priority="", $status="", $planned="", $actual="", $project_id=0)
```

```
Input parameters
$task_id ... task ID
$object_id ... object ID pr path to object (optional)
$to_user ... to_user name (optional)
$startdate ... start date (optional)
$finishdate ... finish date (optional)
$taskname ... name of task (optional)
$message ... task message/description (optional)
$sendmail ... sendmail [true/false]
```

\$priority ... priority [high, medium, low] (optional)
\$status ... status in percent [0-100] (optional)
\$planned ... planned effort in taskunit (optional)
\$actual ... actual effort in taskunit (optional)
\$project_id ... project/subproject ID the task belongs to (optional)

global input parameters

\$mgmt_config
\$hcms_lang_codepage
\$hcms_lang
\$lang

Output

result array

Description

Saves data of a user task and send optional e-mail to user. Since verion 5.8.4 the data will be stored in RDBMS instead of XML files.

7.15.3 deletetask

Syntax

deletetask (\$task_id)

Input parameters

\$task_id ... task ID or array of task IDs to be deleted

global input parameters

\$mgmt_config
\$hcms_lang
\$lang

Output

true/false

Description

Deletes user tasks.

7.15.4 tasknotification

Syntax

tasknotification (\$date)

Input parameters

\$date ... date

global input parameters

\$mgmt_config
\$hcms_lang_codepage
\$hcms_lang
\$lang

Output

true/false

Description

Sends e-mail notifications to users if a task starts or ends on the given date.

7.16 Workflow API Functions

7.16.1 deleteworkflow

Syntax

deleteworkflow (\$site, \$wf_name, \$cat)

Input parameters

\$site ... publication name
\$wf_name ... worfklow name
\$cat ... category [man, script]

global input parameters

\$user \$eventsystem \$mgmt_config \$hcms_lang \$lang

Output

result array

Description

This function deletes a workflow.

7.16.2 buildworkflow

Syntax

buildworkflow (\$workflow_data)

Input parameters

\$workflow_data ... workflow [2 dim. Array]

Output

workflow item

7.16.3 workflowaccept

Syntax

workflowaccept (\$site, \$location, \$object, \$workflow, \$item_id, \$user, \$message, \$sendmail=true, \$priority="medium")

Input parameters

\$site ... publication name [string] \$location ... location name [string] \$object ... object name [string] \$workflow ... workflow [XML-string] \$item_id ... item id [string] \$user ... user name [string] \$message ... task message [string] \$sendmail ... sendmail [true, false] \$priority ... priority [high, medium, low]

global input parameters

\$mgmt_config \$hcms_lang_codepage \$hcms_lang

\$lang

Output

workflow [XML-string]/false

7.16.4 acceptobject

Syntax

acceptobject (\$site, \$location, \$object, \$item_id, \$user, \$message, \$sendmail, \$priority="medium")

Input parameters

\$site ... publication name [string]
\$location ... location name [string]
\$object ... object name [string]
\$item_id ... workflow item id [string]
\$user ... user name [string]
\$message ... task message [string]
\$sendmail ... sendmail [true, false]
\$priority ... priority [high, medium, low]

global input parameters

\$mgmt_config
\$contentfile
\$hcms_lang_codepage
\$hcms_lang
\$lang

Output

array/false

7.16.5 workflowreject

Syntax

workflowreject (\$site, \$location, \$object, \$workflow, \$item_id, \$user, \$message, \$sendmail, \$priority="medium")

Input parameters

\$site ... publication name [string] \$location ... location name [string] \$object ... object name [string] \$workflow ... workflow [XML-string] \$item_id ... workflow item id [string] \$user ... user name [string] \$message ... task message [string] \$sendmail ... send mail [true, false] \$priority ... priority[high, medium, low]

global input parameters

\$mgmt_config
\$hcms_lang
\$lang

Output

workflow [XML-string]/false

7.16.6 rejectobject

Syntax

rejectobject (\$site, \$location, \$object, \$item_id, \$user, \$message, \$sendmail, \$priority="medium")

Input parameters

\$site ... publication name [string] \$location ... location name [string] \$object ... object name [string] \$item_id ... workflow item id [string] \$user ... user name [string] \$message ... task message [string] \$sendmail ... send mail [true, false] \$priority ... priority[high, medium, low]

global input parameters

\$mgmt_config \$contentfile \$hcms_lang \$lang

Output

array/false

7.17 Cloud Storage API Functions

7.17.1 loadcloudobject

Syntax

loadcloudobject (\$site, \$location, \$file, \$user="", \$save=true, \$directoutput=false)

Input parameters

\$site ... publication name \$location ... media file location \$file ... media file name \$user ... user name (optional)

\$save ... save file in local media repository or return content as result [true, false] (optional) \$directoutput ... direct output of the streamed content [true, false] (optional)

global input parameters

\$mgmt_config

Output

data / false on error

Description

This functions load/streams an object/file from the cloud storage and saves it the local media repository on request

7.17.2 savecloudobject

Syntax

savecloudobject (\$site, \$location, \$file, \$user="")

Input parameters

\$site ... publication name

\$location ... media file location \$file ... media file name \$user ... user name (optional)

global input parameters

\$mgmt_config

Output

true / false on error

Description

This function saves an object/file in the cloud storage if no daily cloud snyc has been defined, see main configuration setting \$mgmt_config['storage_dailycloudsnyc']

7.17.3 copycloudobject

Syntax

copycloudobject (\$site, \$location, \$file_old, \$file_new, \$user="")

Input parameters

\$site ... publication name \$location ... media file location \$file_old ... source media file name \$file_new ... destination media file name \$user ... user name (optional)

global input parameters

\$mgmt_config

Output

true / false on error

Description

This functions renames an object/file in the cloud storage

7.17.4 renamecloudobject

Syntax

renamecloudobject (\$site, \$location, \$file_old, \$file_new, \$user="")

Input parameters

\$site ... publication name \$location ... media file location \$file_old ... old media file name \$file_new ... new media file name \$user ... user name (optional)

global input parameters

\$mgmt_config

Output

true / false on error

Description

This functions renames an object/file in the cloud storage

7.17.5 deletecloudobject

Syntax

deletecloudobject (\$site, \$location, \$file, \$user="")

Input parameters

\$site ... publication name \$location ... media file location \$file ... media file name \$user ... user name (optional)

global input parameters

\$mgmt_config

Output

true / false on error

Description

This function deletes an object/file and its data from the cloud storage

7.17.6 synccloudobjects

Syntax

synccloudobjects (\$user="")

Input parameters

\$user ... user name (optional)

global input parameters

\$mgmt_config

Output

true / false on error

Description

This function snychronizes all local objects/files with the cloud storage. See also main configuration settings \$mgmt_config['storage_dailycloudsnyc'] and \$mgmt_config['storage_type'].

7.18 Import/Export API Functions

7.18.1 exportobjects

Syntax

exportobjects ()

Input parameters

global input parameters

\$mgmt_config

Output

true / false

Description

Executes the export job.

7.18.2 importobjects

Syntax

importobjects ()

Input parameters

global input parameters

\$mgmt_config

Output

true / false

Description

Executes the import job.

8 Rechtliche Hinweise / Impressum

8.1 Fragen und Anregungen

Sollten Sie weitergehende Fragen oder Anregungen zum Produkt haben, so wenden Sie sich bitte an den Support. Wir stehen Ihnen auch gerne für Fragen bezüglich unseres Reseller-Programms und Partner-Programms zur Verfügung. Zugriff auf die erweitere Online-Demo des hyper Content & Digital Asset Management Servers können sie ebenfalls über den Support beantragen.

hyperCMS Support:

support@hypercms.com http://www.hypercms.com

8.2 Impressum

Verantwortlich für den Inhalt:

hyperCMS Content Management Solutions GmbH Rembrandtstr. 35/6 A-1020 Wien – Austria

office@hypercms.com http://www.hypercms.com

8.3 Rechtliche Hinweise

Vorliegendes Benutzerhandbuch basiert auf der zum Zeitpunkt der Verfassung des Dokumentes verfügbaren Programmversion.

Der Hersteller behält sich Programmänderungen und -Verbesserungen vor.

Fehler und Irrtümer vorbehalten.

© 2018 by hyperCMS Content Management Solutions