hyper
☐ Content & Digital Asset Management Server
☐ CMS

Version 8.0 Programers Guide

# Index

1	Introd	duction	1
2	hyper	CMS XML-Content-Repository	1
		nyperCMS specific information	
		Neta-Information	
		ext	
		Media	
		inks	
		Components	
_		Articles	
3		ion libraries	
		ncluding a library	
		oading the configuration	
	3.2.1	3	
	3.2.2	Publication Server	8
	3.3	Global variables	8
	3.4 T	emplate variables	.10
	3.5 C	Object operation library	.12
	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	
	3.5.9	copyconnectedobject	
	3.5.10	1	
	3.5.1	g	
	3.5.12	J	
	3.5.13	1	
	3.5.14	4 unpublishobject	.25
	3.5.15	5 getlinkedobject	.26
	3.5.16	6 getconnectedobject	.26
	3.5.17	getobjectcontainer	.27
	3.6 E	dit content	
		ile operation library	
	3.7.1	,	
	3.7.2	savefile	
	3.7.3	loadlockfile	
	3.7.4	savelockfile	
	3.7.4	lockfile	
	3.7.6	unlockfile	
	3.7.7	deletefile	
	3.7.8	appendfile	
		(ML library	
	3.8.1	setxmlparameter	
	3.8.2	getcontentgetcontent	
	3.8.3	getxmlcontent	
	3.8.4	selectcontent	
	3.8.5	selectxmlcontent	.34
	3.8.6	deletecontent	.35
	3.8.7	setcontentsetcontent	36
	3.8.8	updatecontent	.37
	3.8.9	insertcontent	
			.39

		eta Data Generator library	
	3.9.1	getmetakeywords	
	3.9.2	getmetadescription	
		otifications library	
	3.10.1	licensenotification	
4		nents and applications	
5		se Connectivity	
6		eating a Database Connectivity	
7		MS API Function Reference	
/	<i>J</i> 1	ain API Functions	
	7.1 IVIA	correctnumber	
	7.1.1	cleancontent	
	7.1.2	remove_utf8_bom	
	7.1.3	convert_utf8	
	7.1.5	convertchars	
	7.1.6	specialchr	
	7.1.7	specialchr_encode	
	7.1.8	specialchr_decode	
	7.1.9	convertdate	
	7.1.10	offsettime	
	7.1.11	object_exists	
	7.1.12	is_utf8	
	7.1.13	is_latin1	51
	7.1.14	makestring	51
	7.1.15	splitstring	51
	7.1.16	is_folder	51
	7.1.17	is_emptyfolder	52
	7.1.18	is_supported	52
	7.1.19	is_cloudstorage	
	7.1.20	is_cloudobject	
	7.1.21	is_date	
	7.1.22	is_tempfile	
	7.1.23	is_keyword	
	7.1.24	is_thumbnail	
	7.1.25	is_config	
	7.1.26	is_encryptedfile	
	7.1.27	is_document	
	7.1.28	is_image	
	7.1.29	is_rawimage	
	7.1.30	is_alimage	
	7.1.31	is_video	
	7.1.32 7.1.33	is_rawvideois_audio	
	7.1.33	is_mobilebrowser	
	7.1.34	is_iOS	
	7.1.35	is_activelanguage	
	7.1.37	copyrecursive	
	7.1.37	array_iunique	
	7.1.39	in_array_substr	
	7.1.39	createfilename	
	7.1.40	correctfile	
	7.1.41	correctpath	
	7.1.42	convertpath	
	7.1.44	convertlink	
	7.1.45	deconvertpath	
	7.1.46	deconvertlink	
	7.1.47	mediapublicaccess	

		. –
7.1.48	createviewlink	
7.1.49	createportallink	
7.1.50	createaccesslink	
7.1.51	createobjectaccesslink	
7.1.52	createwrapperlink	
7.1.53	createdownloadlink	.68
7.1.54	createmultiaccesslink	.69
7.1.55	createmultidownloadlink	.69
7.1.56	cleandomain	.70
7.1.57	fileversion	.71
7.1.58	createversion	.71
7.1.59	rollbackversion	
7.1.60	deleteversion	
7.1.61	deleteversions	
7.1.62	loadfile_header	
7.1.63	loadfile_fast	
7.1.64	loadfile	
7.1.65	loadlockfile	
7.1.66	savefile	
7.1.67	savelockfile	
7.1.68	appendfile	
7.1.69	lockfile	
7.1.09	unlockfile	
7.1.71	deletefile	
7.1.72	restoremediafile	
7.1.73	preparemediafile	
7.1.74	deletemediafiles	
7.1.75	avoidfilecollision	
7.1.76	substr_in_array	
7.1.77	downloadobject	
7.1.78	downloadfile	
7.1.79	loadcontainer	
7.1.80	savecontainer	
7.1.81	checkworkflow	
7.1.82	inherit_db_load	
7.1.83	inherit_db_read	.84
7.1.84	inherit_db_close	.84
7.1.85	inherit_db_save	.85
7.1.86	inherit_db_getparent	.85
7.1.87	inherit_db_getchild	.86
7.1.88	inherit_db_setparent	.86
7.1.89	inherit_db_insertparent	
7.1.90	inherit_db_deleteparent	
7.1.91	createinstance	
7.1.92	editinstance	
7.1.93	deleteinstance	
7.1.94	createpublication	
7.1.95	editpublication	
7.1.96	editpublicationsetting.	
7.1.97	deletepublication	
7.1.97 7.1.98		
	createpersonalizationdeletepersonalization	
7.1.99	deletepersonalization	
7.1.100	loadtemplate	
7.1.101	createtemplate	
7.1.102	edittemplate	
7.1.103	deletetemplate	
7.1.104	createportal	
7.1.105	editportal	.95

7.1.106	deleteportal	96
7.1.107	createuser	96
7.1.108	edituser	97
7.1.109	deleteuser	98
7.1.110	creategroup	98
7.1.111	editgroup	99
7.1.112	deletegroup	101
7.1.113	renamegroupfolder	101
7.1.114	deletegroupfolder	102
7.1.115	renameworkflowfolder	102
7.1.116	deleteworkflowfolder	103
7.1.117	createmediacat	103
7.1.118	renamemediacat	104
7.1.119	deletemediacat	104
7.1.120	uploadtomediacat	105
7.1.121	deletefrommediacat	105
7.1.122	createfolder	106
7.1.123	createfolders	107
7.1.124	collectfolders	107
7.1.125	copyfolders	108
7.1.126	deletefolder	108
7.1.127	renamefolder	109
7.1.128	correctcontainername	110
7.1.129	createobject	110
7.1.130	uploadfile	
7.1.131	createmediaobject	112
7.1.132	createmediaobjects	112
7.1.133	editmediaobject	113
7.1.134	manipulateobject	114
7.1.135	deletemarkobject	115
7.1.136	deleteunmarkobject	115
7.1.137	deleteobject	116
7.1.138	renameobject	117
7.1.139	renamefile	
7.1.140	cutobject	118
7.1.141	copyobject	
7.1.142	copyconnectedobject	119
7.1.143	pasteobject	120
7.1.144	lockobject	120
7.1.145	unlockobject	121
7.1.146	publishobject	121
7.1.147	publishlinkedobject	122
7.1.148	unpublishobject	123
7.1.149	processobjects	123
7.1.150	collectobjects	124
7.1.151	manipulateallobjects	125
7.1.152	createqueueentry	125
7.1.153	savemessage	126
7.1.154	remoteclient	126
7.1.155	HTTP_Post	
7.1.156	HTTP_Get	
7.1.157	HTTP_Proxy	
7.1.158	loadbalancer	
7.1.159	savelog	
7.1.160	loadlog	
7.1.161	deletelog	
7.1.162	debuglog	
7.1.163	notifyusers	131

	7.1.164	sendlicensenotification	1	31
	7.1.165	licensenotification	1	32
	7.1.166	sendresetpassword	1	32
	7.1.167	html_diff		
	7.1.168	createfavorite		
	7.1.169	deletefavorite		
	7.1.170	rewrite_targetURI		
	7.1.171	rewrite_homepage		
	7.1.171	load_csv		
	7.1.173	sendmessage		
	7.1.174	linking_valid		
	7.1.175	linking_objects		
	7.1.176	linking_inscope		
	7.1.177	savecontent		
7.		API Functions		
		getserverload	1	39
	7.2.2	getconfigvalue	1	39
	7.2.3	getsession	1	40
	7.2.4	getrequest	1	40
	7.2.5	getrequest_esc	1	40
	7.2.6	getuserip	1	41
	7.2.7	getobjectlistcells		
	7.2.8	getlanguageoptions		
	7.2.9	getlanguagefile		
	7.2.10	getcodepagegetcodepage		
	7.2.10	getcalendarlang		
	7.2.11	getescapedtext		
	7.2.12	getsearchhistory		
	7.2.14	gettaxonomy_sublevel		
	7.2.15	gettaxonomy_childs		
	7.2.16	gethierarchy_defintion		
	7.2.17	gethierarchy_sublevel		
	7.2.18	getkeywords		
	7.2.19	getmetakeywords		
	7.2.20	getmetadescription		
	7.2.21	getgooglesitemap	1	47
	7.2.22	getlistelements	1	48
	7.2.23	getmetadata	1	48
	7.2.24	getmetadata_multiobjects	1	49
	7.2.25	getmetadata_container	1	49
	7.2.26	getobjectlist	1	50
	7.2.27	getobjectcontainer	1	50
	7.2.28	getcontainer	1	51
	7.2.29	getwallpaper		
	7.2.30	getcontainername		
	7.2.31	getlocationname		
	7.2.32	getthemes		
	7.2.33	getthemelocation		
	7.2.34	getcategory		
	7.2.35	getpublication		
	7.2.36	getlocationgetlocation		
	7.2.37	getobject		
	7.2.38	getmediacontainername		
	7.2.39	getmediafileversion		
	7.2.40	getobjectid		
	7.2.41	getobjectlink		
	7.2.42	getcontainerversions		
	7.2.43	getlocaltemplates	1	57

7.2.44	gettemplates	157
7.2.45	gettemplateversions	158
7.2.46	getfileinfo	158
7.2.47	getobjectinfo	
7.2.48	getfilesize	
7.2.49	getmimetype	
7.2.50	getfiletype	
7.2.51	getpreviewwidth	
7.2.52	getimagecolorkey	
7.2.53		
	getimagecolors	
7.2.54	getmediasize	
7.2.55	getimageinfo	
7.2.56	getpdfinfo	
7.2.57	getvideoinfo	
7.2.58	getbrowserinfo	
7.2.59	getcontentlocation	
7.2.60	getmedialocation	165
7.2.61	getlockedfileinfo	165
7.2.62	getfavorites	166
7.2.63	gethomeboxes	166
7.2.64	getuserboxes	166
7.2.65	getusersonline	167
7.2.66	getchatstate	
7.2.67	getimagelib	
7.2.68	getfilename	
7.2.69	gethypertag	
7.2.70	gethypertagname	
7.2.71	gethtmltag	
7.2.72		
7.2.72	gethtmltags	
	getattribute	
7.2.74	getoption	
7.2.75	getcharset	
7.2.76	getartid	
7.2.77	getelementid	
7.2.78	getfirstkey	
7.2.79	getdirectoryfiles	
7.2.80	getuserinformation	
7.2.81	getworkflowitem	
	t API Functions	
7.3.1	setsession	
7.3.2	settemplate	174
7.3.3	settaxonomy	174
7.3.4	setarticle	175
7.3.5	settext	175
7.3.6	setmedia	176
7.3.7	setpagelink	177
7.3.8	setcomplink	
7.3.9	sethead	
7.3.10	setrelation	
7.3.11	setfilename	
7.3.12	setuserboxes	
7.3.13	setguiview	
	nnect API Functions	
7.4 00	ftp_userlogon	
7.4.1	ftp_userlogout	
7.4.2 7.4.3	•	
7.4.3 7.4.4	ftp_getfile	
	ftp_putfile	
7.4.5	ftp_deletefile	18∠

	7.4.6	ftp_filelist	183
	7.4.7	createsharelink_facebook	
	7.4.8	createsharelink_twitter	
	7.4.9	createsharelink_googleplus	
	7.4.10	createsharelink_linkedin	
	7.4.11	createsharelink_pinterest	
7.		curity API Functions	
	7.5.1	rootpermission	
	7.5.2	globalpermission	
	7.5.3	localpermission	
	7.5.4	accessgeneral	
	7.5.5	accesspermission	
	7.5.6	setlocalpermission	
	7.5.7	checkpublicationpermission	
	7.5.8	checkadminpermission	
	7.5.9	checkrootpermission	
	7.5.10	checkglobalpermission	
	7.5.11	checklocalpermission	
	7.5.12	checklanguage	
	7.5.13	checkgroupaccess	
	7.5.14	userlogin	
	7.5.15	setportalpermissions	
	7.5.16	registerinstance	
	7.5.17	registeruser	
	7.5.17	registerassetbrowser	
	7.5.19	createchecksum	
	7.5.20	writesession	
	7.5.21	writesessiondata	
	7.5.22	createsession	
	7.5.23	killsession	
	7.5.24	checkdiskkey	
	7.5.25	checkpassword	
	7.5.26	loguserip	
	7.5.27	checkuserip	
	7.5.28	checkuserrequests	
	7.5.29	checkusersession	
	7.5.30	allowuserip	
	7.5.30	valid_objectname	
	7.5.32	valid_locationname	
	7.5.32	valid_publicationname	
	7.5.34	html_encode	
	7.5.35	html_decode	
	7.5.36	scriptcode_encode	
	7.5.37	scriptcode_extract	
	7.5.38	scriptcode_clean_functions	
	7.5.39	sql_clean_functions	
	7.5.40	url_encode	
	7.5.41	url_decode	
	7.5.42	shellcmd_encode	
	7.5.42	hcms_crypt	
	7.5.44	hcms_encrypt	
	7.5.45	hcms_decrypt	
	7.5.46	createtimetoken	
	7.5.40	checktimetoken	
	7.5.47	createtoken	
	7.5.49	checktoken	
	7.5.50	createuniquetoken	
	7.5.50	createpassword	
	,		201

7.5.52	rand_secure	207
	dia API Functions	
7.6.1	ocr_extractcontent	
7.6.2	indexcontent	
7.6.2		
,	unindexcontent	
7.6.4	reindexcontent	
7.6.5	createthumbnail_indesign	
7.6.6	createthumbnail_video	211
7.6.7	createimages_video	211
7.6.8	createmedia	212
7.6.9	splitmedia	213
7.6.10	convertmedia	
7.6.11	convertimage	
7.6.12	rotateimage	
7.6.12	hex2rgb	
7.6.13	rgb2hex	
7.6.15	readmediaplayer_config	
7.6.16	savemediaplayer_config	
7.6.17	createdocument	
7.6.18	unzipfile	
7.6.19	clonefolder	
7.6.20	zipfiles_helper	219
7.6.21	zipfiles	220
7.6.22	px2mm	
7.6.23	px2inch	
7.6.24	inch2px	
7.6.25	sec2time	
7.6.26	mediasize2frame	
7.6.27	vtt2array	
7.7 Me	tadata API Functions	222
7.7 Me 7.7.1	tadata API Functionsimportmetadata	222 222
7.7 Me 7.7.1 7.7.2	tadata API Functionsimportmetadataloadtaxonomy	222 222 223
7.7 Me 7.7.1	tadata API Functionsimportmetadata	222 222 223
7.7 Me 7.7.1 7.7.2	tadata API Functionsimportmetadataloadtaxonomy	222 222 223 224
7.7 Me 7.7.1 7.7.2 7.7.3	tadata API Functionsimportmetadataloadtaxonomysavetaxonomy	222 222 223 224 224
7.7 Me 7.7.1 7.7.2 7.7.3 7.7.4	tadata API Functions. importmetadata loadtaxonomy. savetaxonomy createtaxonomy splitkeywords	222 222 223 224 224
7.7 Me 7.7.1 7.7.2 7.7.3 7.7.4 7.7.5 7.7.6	tadata API Functions. importmetadata loadtaxonomy savetaxonomy createtaxonomy splitkeywords copymetadata	222 223 224 224 225
7.7 Me 7.7.1 7.7.2 7.7.3 7.7.4 7.7.5 7.7.6 7.7.7	tadata API Functions. importmetadata loadtaxonomy savetaxonomy createtaxonomy splitkeywords copymetadata extractmetadata	222 223 224 224 225 225
7.7 Me 7.7.1 7.7.2 7.7.3 7.7.4 7.7.5 7.7.6 7.7.7 7.7.8	tadata API Functions. importmetadata loadtaxonomy savetaxonomy createtaxonomy splitkeywords copymetadata extractmetadata xmlobject2array	222 223 224 225 225 225
7.7 Me 7.7.1 7.7.2 7.7.3 7.7.4 7.7.5 7.7.6 7.7.7 7.7.8 7.7.9	tadata API Functions. importmetadata loadtaxonomy savetaxonomy createtaxonomy splitkeywords copymetadata extractmetadata xmlobject2array id3_getdata	222223224225225225
7.7 Me 7.7.1 7.7.2 7.7.3 7.7.4 7.7.5 7.7.6 7.7.7 7.7.8 7.7.9 7.7.10	tadata API Functions. importmetadata loadtaxonomy savetaxonomy createtaxonomy splitkeywords copymetadata extractmetadata xmlobject2array id3_getdata id3_writefile	222 223 224 225 225 225 226 226
7.7 Me 7.7.1 7.7.2 7.7.3 7.7.4 7.7.5 7.7.6 7.7.7 7.7.8 7.7.9 7.7.10 7.7.11	tadata API Functions. importmetadata loadtaxonomy savetaxonomy createtaxonomy splitkeywords copymetadata extractmetadata xmlobject2array id3_getdata id3_writefile id3_create	222223224225225225226226
7.7 Me 7.7.1 7.7.2 7.7.3 7.7.4 7.7.5 7.7.6 7.7.7 7.7.8 7.7.9 7.7.10 7.7.11 7.7.12	tadata API Functions. importmetadata loadtaxonomy savetaxonomy createtaxonomy splitkeywords copymetadata extractmetadata xmlobject2array id3_getdata id3_writefile id3_create xmp_getdata	222223224225225225226226227227
7.7 Me 7.7.1 7.7.2 7.7.3 7.7.4 7.7.5 7.7.6 7.7.7 7.7.8 7.7.9 7.7.10 7.7.11 7.7.12 7.7.13	tadata API Functions.  importmetadata loadtaxonomy savetaxonomy createtaxonomy splitkeywords copymetadata extractmetadata xmlobject2array id3_getdata id3_writefile id3_create xmp_getdata xmp_writefile	222 223 224 225 225 225 226 227 227 228
7.7 Me 7.7.1 7.7.2 7.7.3 7.7.4 7.7.5 7.7.6 7.7.7 7.7.8 7.7.9 7.7.10 7.7.11 7.7.12 7.7.13 7.7.14	tadata API Functions.  importmetadata loadtaxonomy savetaxonomy createtaxonomy splitkeywords copymetadata extractmetadata xmlobject2array id3_getdata id3_writefile id3_create xmp_getdata xmp_writefile xmp_create	222 223 224 225 225 225 226 227 227 228 229
7.7 Me 7.7.1 7.7.2 7.7.3 7.7.4 7.7.5 7.7.6 7.7.7 7.7.8 7.7.9 7.7.10 7.7.11 7.7.12 7.7.13	tadata API Functions.  importmetadata loadtaxonomy savetaxonomy createtaxonomy splitkeywords copymetadata extractmetadata xmlobject2array id3_getdata id3_writefile id3_create xmp_getdata xmp_writefile	222 223 224 225 225 225 226 227 227 228 229
7.7 Me 7.7.1 7.7.2 7.7.3 7.7.4 7.7.5 7.7.6 7.7.7 7.7.8 7.7.9 7.7.10 7.7.11 7.7.12 7.7.13 7.7.14	tadata API Functions.  importmetadata loadtaxonomy savetaxonomy createtaxonomy splitkeywords copymetadata extractmetadata xmlobject2array id3_getdata id3_writefile id3_create xmp_getdata xmp_writefile xmp_create	222223224225225225226227227228229
7.7 Me 7.7.1 7.7.2 7.7.3 7.7.4 7.7.5 7.7.6 7.7.7 7.7.8 7.7.9 7.7.10 7.7.11 7.7.12 7.7.13 7.7.14 7.7.15	tadata API Functions. importmetadata loadtaxonomy savetaxonomy createtaxonomy splitkeywords copymetadata extractmetadata xmlobject2array id3_getdata id3_writefile id3_create xmp_getdata xmp_create geo2decimal	222 223 224 225 225 225 226 227 227 228 229 229
7.7 Me 7.7.1 7.7.2 7.7.3 7.7.4 7.7.5 7.7.6 7.7.7 7.7.8 7.7.9 7.7.10 7.7.11 7.7.12 7.7.13 7.7.14 7.7.15 7.7.16	tadata API Functions importmetadata loadtaxonomy savetaxonomy createtaxonomy splitkeywords copymetadata extractmetadata xmlobject2array id3_getdata id3_writefile id3_create xmp_getdata xmp_writefile xmp_create geo2decimal exif_getdata iptc_getdata	222 223 224 225 225 226 227 227 228 229 229 229
7.7 Me 7.7.1 7.7.2 7.7.3 7.7.4 7.7.5 7.7.6 7.7.7 7.7.8 7.7.9 7.7.10 7.7.11 7.7.12 7.7.13 7.7.14 7.7.15 7.7.16 7.7.17	tadata API Functions importmetadata loadtaxonomy savetaxonomy createtaxonomy splitkeywords copymetadata extractmetadata xmlobject2array id3_getdata id3_writefile id3_create xmp_getdata xmp_writefile xmp_create geo2decimal exif_getdata iptc_getdata iptc_getcharset	222 223 224 225 225 225 226 227 228 229 229 230 230
7.7 Me 7.7.1 7.7.2 7.7.3 7.7.4 7.7.5 7.7.6 7.7.7 7.7.8 7.7.9 7.7.10 7.7.11 7.7.12 7.7.13 7.7.14 7.7.15 7.7.16 7.7.17 7.7.18 7.7.18 7.7.19	tadata API Functions. importmetadata loadtaxonomy savetaxonomy createtaxonomy splitkeywords copymetadata extractmetadata xmlobject2array id3_getdata id3_writefile id3_create xmp_getdata xmp_writefile xmp_create geo2decimal exif_getdata iptc_getdata iptc_getcharset iptc_maketag	222223224225225225226227227228229229230231
7.7 Me 7.7.1 7.7.2 7.7.3 7.7.4 7.7.5 7.7.6 7.7.7 7.7.8 7.7.9 7.7.10 7.7.11 7.7.12 7.7.13 7.7.14 7.7.15 7.7.16 7.7.16 7.7.17 7.7.18 7.7.19 7.7.20	tadata API Functions importmetadata loadtaxonomy savetaxonomy createtaxonomy splitkeywords copymetadata extractmetadata xmlobject2array id3_getdata id3_writefile id3_create xmp_getdata xmp_writefile xmp_create geo2decimal exif_getdata iptc_getdata iptc_getcharset iptc_maketag iptc_writefile.	222223224225225225226227227228229229230231231
7.7 Me 7.7.1 7.7.2 7.7.3 7.7.4 7.7.5 7.7.6 7.7.7 7.7.8 7.7.9 7.7.10 7.7.11 7.7.12 7.7.13 7.7.14 7.7.15 7.7.16 7.7.17 7.7.18 7.7.19 7.7.20 7.7.21	tadata API Functions. importmetadata loadtaxonomy. savetaxonomy createtaxonomy splitkeywords copymetadata extractmetadata xmlobject2array id3_getdata id3_writefile id3_create xmp_getdata xmp_writefile xmp_create geo2decimal exif_getdata iptc_getcharset iptc_maketag iptc_writefile iptc_create	222224224225225225226227227228229229230231231
7.7 Me 7.7.1 7.7.2 7.7.3 7.7.4 7.7.5 7.7.6 7.7.7 7.7.8 7.7.9 7.7.10 7.7.11 7.7.12 7.7.13 7.7.14 7.7.15 7.7.16 7.7.17 7.7.18 7.7.19 7.7.20 7.7.21 7.7.22	tadata API Functions. importmetadata loadtaxonomy savetaxonomy createtaxonomy splitkeywords copymetadata extractmetadata xmlobject2array id3_getdata id3_writefile id3_create xmp_getdata xmp_writefile xmp_create geo2decimal exif_getdata iptc_getdata iptc_getcharset iptc_maketag iptc_writefile iptc_create createmapping	222 223 224 225 225 225 226 227 227 228 229 229 230 231 231 232 232
7.7 Me 7.7.1 7.7.2 7.7.3 7.7.4 7.7.5 7.7.6 7.7.7 7.7.8 7.7.9 7.7.10 7.7.11 7.7.12 7.7.13 7.7.14 7.7.15 7.7.16 7.7.17 7.7.18 7.7.19 7.7.20 7.7.21 7.7.22 7.7.23	tadata API Functions importmetadata loadtaxonomy savetaxonomy createtaxonomy splitkeywords copymetadata extractmetadata xmlobject2array id3_getdata id3_writefile id3_create xmp_getdata xmp_writefile xmp_create geo2decimal exif_getdata iptc_getdata iptc_getcharset iptc_maketag iptc_writefile iptc_create createmapping getmapping	222223224225225226227227228229229229230231231232
7.7 Me 7.7.1 7.7.2 7.7.3 7.7.4 7.7.5 7.7.6 7.7.7 7.7.8 7.7.9 7.7.10 7.7.11 7.7.12 7.7.13 7.7.14 7.7.15 7.7.16 7.7.17 7.7.18 7.7.18 7.7.19 7.7.20 7.7.21 7.7.22 7.7.23 7.7.24	tadata API Functions importmetadata loadtaxonomy savetaxonomy createtaxonomy splitkeywords copymetadata extractmetadata xmlobject2array id3_getdata id3_writefile id3_create xmp_getdata xmp_writefile xmp_create geo2decimal exif_getdata iptc_getdata iptc_getcharset iptc_maketag iptc_writefile iptc_create createmapping getmapping setmetadata	222223224225225226227227227228229229230231231232233
7.7 Me 7.7.1 7.7.2 7.7.3 7.7.4 7.7.5 7.7.6 7.7.7 7.7.8 7.7.9 7.7.10 7.7.11 7.7.12 7.7.13 7.7.14 7.7.15 7.7.16 7.7.17 7.7.18 7.7.18 7.7.19 7.7.20 7.7.21 7.7.22 7.7.23 7.7.24 7.8 Lin	tadata API Functions importmetadata loadtaxonomy savetaxonomy createtaxonomy splitkeywords copymetadata extractmetadata xmlobject2array id3_getdata id3_writefile id3_create xmp_getdata xmp_writefile xmp_create geo2decimal exif_getdata iptc_getdata iptc_getcharset iptc_maketag iptc_writefile iptc_create createmapping getmapping setmetadata k API Functions	222223224225225225226227227228229229230231231232233233
7.7 Me 7.7.1 7.7.2 7.7.3 7.7.4 7.7.5 7.7.6 7.7.7 7.7.8 7.7.9 7.7.10 7.7.11 7.7.12 7.7.13 7.7.14 7.7.15 7.7.16 7.7.17 7.7.18 7.7.19 7.7.20 7.7.21 7.7.22 7.7.23 7.7.24 7.8 Lin 7.8.1	tadata API Functions. importmetadata loadtaxonomy savetaxonomy createtaxonomy splitkeywords copymetadata extractmetadata xmlobject2array id3_getdata id3_writefile id3_create xmp_getdata xmp_writefile xmp_create geo2decimal exif_getdata iptc_getdata iptc_getdata iptc_getdata iptc_maketag iptc_writefile iptc_create createmapping getmapping setmetadata k API Functions link_db_restore	222223224225225225226227227228229229230231231231232233233
7.7 Me 7.7.1 7.7.2 7.7.3 7.7.4 7.7.5 7.7.6 7.7.7 7.7.8 7.7.9 7.7.10 7.7.11 7.7.12 7.7.13 7.7.14 7.7.15 7.7.16 7.7.17 7.7.18 7.7.18 7.7.19 7.7.20 7.7.21 7.7.22 7.7.23 7.7.24 7.8 Lin	tadata API Functions importmetadata loadtaxonomy savetaxonomy createtaxonomy splitkeywords copymetadata extractmetadata xmlobject2array id3_getdata id3_writefile id3_create xmp_getdata xmp_writefile xmp_create geo2decimal exif_getdata iptc_getdata iptc_getcharset iptc_maketag iptc_writefile iptc_create createmapping getmapping setmetadata k API Functions	222223224225225225226227227228229229230231231231232233233

7.8.4	link_db_close	
7.8.5	link_db_save	
7.8.6	link_db_update	
7.8.7	link_db_insert	
7.8.8 7.8.9	link_db_getobjectlink_db_getobject	
7.8.10	link_update	
7.8.10	getlinkedobject	
7.8.12	getconnectedobject	
7.8.13	extractlinks	
	gin API Functions	
7.9.1	plugin_getdefaultconf	
7.9.2	plugin_readmenu	240
7.9.3	plugin_parse	241
7.9.4	plugin_generatedefinition	241
7.9.5	plugin_saveconfig	
7.9.6	plugin_generatelink	
	er Interface API Functions	
7.10.1	windowwidth	
7.10.2	windowheight	
7.10.3	toggleview	
7.10.4	togglesidebar	
7.10.5	setfilter	
7.10.6	objectfilter	
7.10.7 7.10.8	showdate	
7.10.6	showshorttextshowtopbar	
7.10.9	showtoppalshowtopmenubar	
7.10.10	showhomeboxes	
7.10.11	showmessage	
7.10.13	showinfopage	
7.10.14	showinfobox	
7.10.15	showsharelinks	
7.10.16	showmetadata	
7.10.17	showobject	251
7.10.18	showmedia	251
7.10.19	showcompexplorer	252
7.10.20	showeditor	
7.10.21	showinlineeditor_head	
7.10.22	showinlinedatepicker_head	
7.10.23	showinlineeditor	
7.10.24	showvideoplayer	
7.10.25	showvideoplayer_head	
7.10.26	showaudioplayer	
7.10.27	showaudioplayer_head	
7.10.28 7.10.29	debug_getbacktracestringshowAPIdocs	
7.10.29	readnavigation	
7.10.30	createnavigation	
7.10.31	shownavigation	
7.10.32	showselect	
7.10.34	showtranslator	
7.10.35	showmapping	
7.10.36	showgallery	
7.10.37	showthumbnail	
7.10.38	showworkflowstatus	
7.11 Ter	nplate Engine API Functions	264
7.11.1	transformlink	264

7.11.2	followlink	264
7.11.3	errorhandler	
7.11.4	viewinclusions	
7.11.5	buildview	
7.11.6	unescapeview	
	•	
7.11.7	buildsearchform	
7.11.8	buildbarchart	
	API Functions	
7.12.1	valid_tagname	
7.12.2	setxmlparameter	
7.12.3	getcontent	
7.12.4	geticontent	.270
7.12.5	getxmlcontent	.270
7.12.6	getxmlicontent	.271
7.12.7	selectcontent	.271
7.12.8	selecticontent	
7.12.9	selectxmlcontent	
7.12.10	selectxmlicontent	
7.12.11	deletecontent	
7.12.11	deleteicontent	
7.12.12	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	.278
7.12.20	addicontent	.278
7.13 Repo	ort API Functions	.279
7.13.1	createreport	.279
7.13.2	editreport	.280
7.13.3	loadreport	
7.13.4	deletereport	
7.13.5	analyzeSQLselect	
7.13.6	showpiechart	
7.13.7	showcolumnchart	
	showareachartshowareachart	
7.13.8		
7.13.9	showtimelinechart	
7.13.10	showgeolocationchart	
-	ect API Functions	
7.14.1	createproject	
7.14.2	editproject	
7.14.3	deleteproject	
7.14.4	getprojectstructure	
7.14.5	showprojectrecordshowprojectrecord	. 287
7.14.6	showganttchartshowganttchart	.288
7.15 Task	API Functions	.289
7.15.1	createtask	.289
7.15.2	edittask	
7.15.3	deletetask	
7.15.4	tasknotification	
	kflow API Functions	
7.16 Work	deleteworkflow	
7.16.1 7.16.2		
7.16.2 7.16.3	buildworkflow	
	workflowaccept	
7.16.4	acceptobject	
7.16.5	workflowreject	
7.16.6	rejectobject	. 294

	7.17 C	Cloud Storage API Functions	295
	7.17.1		
	7.17.2		
	7.17.3	3 copycloudobject	296
	7.17.4	renamecloudobject	296
	7.17.5	deletecloudobject	297
	7.17.6	synccloudobjects	297
	7.17.7	7 GCanalyzevideo	298
	7.17.8	GCanalyzeimage	298
	7.17.9	GCspeech2text	298
	7.18 In	mport/Export API Functions	
	7.18.1	l exportobjects	299
	7.18.2	2 importobjects	300
	7.18.3		
	7.18.4	1 importxmlfiles	301
	7.18.5	5 calculatedate	301
	7.18.6	s exportxmlfiles	301
8	Legal	reference / flag	303
	8.1 C	Questions and suggestions	303
	8.2 In	mprint	303
	8.3 L	egal information	303

# 1 Introduction

The following chapters deal with the function libraries of the hyper Content & Digital Asset Management Server and thus provide the documentation of the API (Application Programming Interface).

All libraries are located within the hyperCMS installation in the folder "function" and can be integrated and used in the respective scripts or templates. This can be used, for example, to create dynamic pages (applications) using the XML content repository.

If you run your application on a physically separated server, it is important that the function libraries are also available on the publication server. In this case you need to have access to the corresponding files on the publication server as well.

# 2 hyperCMS XML-Content-Repository

The XML content repository includes all XML Content Container and thus provides all content in native XML. The structure (schema) within an XML content container is dynamically generated based on the template used and has the following appearance:

```
<?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>fgfdgfdg</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>
 linkcollection>
  link >
    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>
```

After a review of the content container, a structure can be seen, which is composed of the following main elements for content storage:

- hyperCMS specific information
- Meta-information
- Text
- Media (images or other multimedia files)
- Links
- Components
- Articles

The entire content is made up of these basic elements whose information is stored within XML tags.

Articles include the elements text, media and links as well. The entire contents of a page or component can be obtained from the associated content containers.

# 2.1 hyperCMS specific information

The data collected in this XML node represent primarily relevant information for the management of the container.

# **Description:**

Name of the Content Container (unique for all publications) Schema of the object: page (page) or component (comp)
Object (page or component) that led tot he creation of the Content
Containers
All objects which use the Container
Object owner (user)
Date of the last changes of the Containers
Date of the last publishing of the object based on the Content Container
Status can be "active" if an object using the Cotainer exists. If all objects
using the Container have been removed, the status will be set to
"deleted". The Container therefore holds the last published information,

but it can not be used anymore.

# 2.2 Meta-Information

The standard meta-information of a HTML page is described in this XML node.

```
<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
```

#### **Description:**

pagetitle Page title pageauthor Author

pagedescription Description of the content of a page

pagekeywords List of keywords oft he page

pagecontenttype Content-Type (character set) of the page or component

pagelanguage Language shortcut of the page pagerevisit Search engine revisit of the page

# 2.3 Text

This XML-node stores the text.

<text>
<text\_id>headline</text\_id>
<textuser>demouser</textuser>
<textcontent>fgfdgfdg</textcontent>
</text>

# **Description:**

text\_id Text identification

textuser Text owner (last changes of the Text by a user)

textcontent text content

# 2.4 Media

This XML-node describes an included media file.

```
<media>
<media_id>logo</media_id>
<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>
```

# **Description:**

media\_id Media identification

mediauser Media owner (last changes of the Media by a user)

mediafile included multimedia file and declaration of the Publication

mediaobject Location of the multimedia file
mediaaltext Alternative text of the multimedia file
mediaalign Alignment of the multimedia file
medawidth Displayed width of the multimedia file
mediaheight Displayed height of the multimedia file

# 2.5 Links

This XML-node describes the link to a page or file.

```
<link>
<link_id>link</link_id>
<linkuser>demouser</linkuser>
<linkhref>http://localhost/index.php</linkhref>
<linktarget>_blank</linktarget>
<linktext>click me</linktext>
</link></link>
```

# **Description:**

link id Link identification

linkuser Link owner (last changes of the Link by a user)

linkhref Reference (Link) to a page or file

linktarget Target of the reference (name of the target frame)

linktext Text describing the link

# 2.6 Components

This XML-node describes the reference to Components.

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

### **Description:**

component\_id Component identification

componentuser Component owner (last changes of the Component reference by a user)

componentcond assigned customer Profile to the Component

componentfiles Reference (Component-link) to a single or multiple Components

# 2.7 Articles

This XML-node describes the article information.

```
<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>
```

#### **Description:**

article\_id Article identification articletitle Title of the Article

articeldatefrom
articeldateto
articlestatus

Publishing date of the Article
End date of publishing the Article
Publishing settings of the Article:

active = always published/displaye

active = always published/displayed inactive = never published/displayed

timeswitched = scheduled publishing/display

articleuser Article owner (last changes of the Article by a user)

articlecollection Holds all content of the Article

# 3 Function libraries

# 3.1 Including a library

The inclusion of a configuration or library requires that you know the absolute or relative path to the library. By using the function "require" or "require\_once" and specifying the path to the library file all functions contained in the library will be available. Once the library is included, all functions can be used in the script.

To use the hyperCMS functions, the file "hypercms\_api.inc.php" needs to be included. This file contains all functions required for programming.

```
// absolute path on MS Windows
require_once ("C:/inetpub/wwwroot/hypercms/function/hypercms_api.inc.php");
// relative path on MS Windows or Linux/UNIX
require_once ("function/hypercms_api.inc.php");
```

# 3.2 Loading the configuration

# 3.2.1 Content Management Server

To use the main configuration of hyperCMS the appropriate configuration file must be loaded. The main configuration will be loaded when including the hyperCMS API. However you can also load it in your script.

Using the variable \$site for the identification of a publication, the publication can be loaded as well. The hyperCMS config file is located in "hypercms/config" and is named "config.inc.php". The publication config files are located in hyperCMS Data directory in the directory "data/config". Its filename holds the name of the publication as well as the ending "inc.php.", example: site.inc.php.

```
// Inlcude the main config file (please set the correct path):
require_once ("C:/inetpub/wwwroot/hypercms/config.inc.php");

// Include publication management config file
// Attention: Please use valid_publicationname to verify the name before including the file
if (valid_publicationname ($site))
{
    require_once ($mgmt_config['abs_path_data']."config/".$site.".conf.php");
}
```

The config files can be opened and read. Each parameter is described therein and is available for use in programs. Therefore, please take a look at the configuration to learn more about the parameters and their names.

If you want to set a specific language language, the variable \$lang need to be set. \$lang contains the language code, which is defined in the main configuration file "hypercms/config/config.inc.php".

```
// Set the language for messages in functions, German (de)
$lang = "de";
```

Since you want to use the hyperCMS API you need to include the hyperCMS API loader.

```
// Include the hyperCMS API: require_once ($mgmt_config['abs_path_cms']."/function/hypercms_api.inc.php");
```

Now you can start using the API functions. For instance loading the content container of an object using various methods:

```
// Loading the page $pagedata = loadfile ("%page%/MyPublication/home/", "index.php");

// Reading the name of the content container $contentcontainer = filepointer ($pagedata, "content");

// Loading the live content container from the content repository $containerdata = loadcontainer ($contentcontainer, "published", $user);

// Or even more simple by using the direct path to the object $containerdata = getobjectcontainer ("MyPublication", "%page%/MyPublication/home/", "index.php", $user);
```

The functions will also load the publication specific configuration in case it is not provided. Since many features require the settings of a publication, it is advisable to include the configuration before you plan any actions.

#### 3.2.2 Publication Server

Note that the configuration of the publication server (publication target) is stored separately in an INI file. If you will need the publication target settings, you must load and parse the INI file. After that you can access the settings as an array.

The INI file of the publication target is located in the external repository in the directory "repository/config". The file name corresponds to the name of the publication with the file extension ".ini".

```
// Load and parse the INI file using PHP
$publ_config = parse_ini_file ("C:/inetpub/wwwroot/repository/config/Mandant_1.ini");
// Access the settings oft he publication target
echo "This is the document root of the publication: ".$publ_config[abs_publ_page];
```

# 3.3 Global variables

Many functions use global variables that are stored in the configuration and are available to functions as global. You should therefore take care that those global variable names of hyperCMS are not changed in your scripts.

The following list shows all global variables of hyperCMS, which must not be changed with in your own scripts:

\$mgmt\_config \$lang \$lang\_name \$lang\_shortcut \$lang\_codepage \$lang\_shortcut\_default Many global variables of hyperCMS are useful for use in hyperCMS scripts and PHP scripts, these are only available if the corresponding configuration has been loaded, or a hyperCMS script (used only during the publication process) is in use. Since this happened in the preview as well when publishing pages and components, these variables can be used in hyperCMS scripts. For dynamic applications that are executed each time a visitor accesses a page or component, the configuration must be integrated directly in the template, if hyperCMs variables are required.

# **Content Management Server:**

language shortcut according to config.inc.php \$lang \$mgmt\_config['url\_path\_cms'] URL of the hyperCMS root directory according to

config.inc.php

**\$mgmt\_config[**'abs\_path\_cms'] absolute path to the hyperCMS root directory according to

config.inc.php

\$mgmt\_config['url\_path\_page'] URL of the document root of the publication in the

management system

\$mgmt\_config['abs\_path\_page'] absolute of the document root of the publication in the

management system

**\$mgmt\_config[**'url\_path\_comp'] URL of the component root directory of the publication in

the management system

\$mgmt\_config['abs\_path\_comp']absolute path of the component root directory of the

publication in the management system

#### **Publication Server:**

hyperCMS scripts can access variables at any time. The values are stored in the array \$publ\_config, but are also optionally available without the array. If the script/application will be executed at each access of a page or component on the publication target, the configuration file must be loaded separately.

URL of the document root of the publication target **\$publ\_config[**'url\_publ\_page'] **\$publ\_config[**'abs\_publ\_page'] absolute path of the document root of the publication

**\$publ\_config[**'url\_publ\_comp'] URL of the document root of the publication target **\$publ\_config[**'abs\_publ\_comp'] absolute path of the document root of the publication

target

Optional (deprected):

\$url\_publ\_page URL of the document root of the publication target \$abs\_publ\_page absolute path of the document root of the publication

\$url\_publ\_comp URL of the document root of the publication target absolute path of the document root of the publication \$abs\_publ\_comp

target

# 3.4 Template variables

There is also the possibility to use hyperCMS template variables in templates. These variables are a special feature, since they don not need to be used in hyperCMS script. Rather, they are placeholder for the value of a variable and can be used in any template.

This neutral form of the variables should primarily be used in templates, providing a more technology-neutral usage.

Please pay attention to the lower case of all variables!

**%container%** for the name of the content container of an object **%container\_id%** for the ID of the content container of an object

**%objecthash%** for the hash of an object **%object\_id%** for the ID of an object

%template% for the template file name of an object used %publication% for the publication where the object is located

**%url\_location%** for the absolute path (URL) of the location where the object is located **%abs\_location%** for the absolute path in the filesystem of the location where the object is

located

**%object%** for the name of the object

%date% describes the current date (format: JJJJ-MM-TT)

**%view%** describes the display mode:

publish ... published

cmsview ... Editing view in EasyEdit mode

preview ... Preview

formedit ... Editing mode in form view formlock ... form view with editing locked formmeta ... Metadata in form view template ... Template preview

To integrate media files (assets) we can make use of a path variable. When publishing a page or component, the path variable is for instance replaced by the URL (address) of the target of the publication:

%tplmedia% for the absolute path (URL) of the template media repository

%url\_media% for the absolute path (URL) of the content media repository (Alternative

%media% can be used)

**%abs\_media%** for the absolute path in the filesystem of the content media repository

The document root of pages and components of the publication target can be provided as well:

**%url\_page%** for the absolute path (URL) of the page root folder

**%abs\_page%** for the absolute path in the filesystem of the page root folder **%url\_comp%** for the absolute path (URL) of the component root folder

%abs\_comp% for the absolute path in the filesystem of the component root folder
%url\_rep% for the absolute path (URL) of the external repository root
%abs\_rep% for the absolute path in the filesystem of the external repository root

folder

%url\_ hypercms% for the absolute path (URL) of the hyperCMS root folder

%abs\_hypercms% for the absolute path in the filesystem of the hyperCMS root folder

Please keep in mind to add a slash "/" to the end of the path variable, if you would like to complement the variable with a continuative path.

#### Definition of the date format when the format-attribute is used in the textd-Tag:

```
%a
      'am' or 'pm'
%A
      'AM' or 'PM'
%d
      day of the month, 2 digits with leading zeroes (01 to 31)
%D
      day of the week, textual, short, eg "Fri"
%F
      month, textual, long; eg "January"
%h
      hour, 12-hour format (01 to 12)
%Н
      hour, 24-hour format (00 to 23)
%g
      hour, 12-hour format without leading zeros (1 to 12)
%G
      hour, 24-hour format without leading zeros (0 to 23)
%i
      minutes (00 to 59)
%j
      day of the month without leading zeros (1 to 31)
%l
      day of the week, textual, long, eg "Friday"
      1 if leap year, otherwise - 0
%L
%m
      month (01 to 12)
%n
      month without leading zeros (1 to 12)
%M
      month, textual, short, eg "Jan"
%s
      seconds (00 to 59)
      number of days in the month (28 to 31)
%t
      day of the week, numeric (0, Sunday to 6, Saturday)
%w
%Y
      year, 4 digits, eg 2007
%у
      year, 2 digits, eg "07"
%z
      day of the year (1 to 366)
```

If you are using the hyperCMS APIs, it is often advisable to use the place holders %page% and %comp% to access the document root of pages and components. This path variables can be used only on the management side.

It should be noted that the variable is always paired with the publication name to form the root directory, eg:

%page%/%publication%/ .... Pages document root of the current publication

**%page%/Publikationsname/** provides the path to root directory of the pages document root.

**%comp%/Publikationsname/** provides the path to root directory of the components document root.

# 3.5 Object operation library

This library contains all functions for the manipulation of objects (pages, components or files). You should only use these functions to access objects that are managed by the system.

#### 3.5.1 createfolder

### Syntax:

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

#### **Description:**

Creates a new folder.

Example:

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

Input-Parameters:

\$site Name of the publication

\$location absolute path (location of the new folder)

\$foldernew Name of the new folder

\$user User name

# globale Input-Parameters:

The following global input parameters need to be passed to the function:

\$lang Language setting or language shortcut, e.g. "en", "de"

**Output:** 

Array \$result holds the following information:

\$result[result] True/False (has the folder been created successfully)

\$result[add\_onload] JavaScript code for the onLoad event

\$result[message] Message regarding the result of the action or error message

\$result[folder] Name of the folder

#### 3.5.2 deletefolder

#### Syntax:

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

#### Description:

Removes an existing folder. The folder is removed only if it contains no more objects. All objects must therefore be removed by using the function deleteobject.

#### Example:

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

#### Input parameters:

\$site Name of the publication

\$location absolute path (location of the folder)

\$folder Name of the folder

\$user User name

# global input parameters:

The following global input parameters need to be passed to the function:

\$lang Language setting or language shortcut, e.g. "en", "de"

# Output:

Array \$result holds the following information:

\$result[result] True/False (has the folder been removed successfully)

\$result[add\_onload] JavaScript code for the onLoad event

\$result[message] Message regarding the result of the action or error message \$result[folder] Name of the existing folder is not successful, otherwise empty

#### 3.5.3 renamefolder

#### Syntax:

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

# **Description:**

Renames an existing folder.

Example:

\$result = renamefolder ("%publication%", "%page%/%publication%/", "company", "news",
"Miller");

# Input parameters:

\$site Name of the publication

\$location absolute path (location of the folder)

\$folder old folder name \$foldernew new folder name \$user User name

### global input parameters:

The following global input parameters need to be passed to the function:

\$lang Language setting or language shortcut, e.g. "en", "de"

# Output:

Array \$result holds the following information:

\$result[result] True/False (Could the folder be renamed successfully)

\$result[add\_onload] JavaScript code for the onLoad event

\$result[message] Message regarding the result of the action or error message

\$result[folder] Name of the folder

# 3.5.4 createobject

#### Syntax:

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

#### **Description:**

Creates a new page or component based on a template. Please note that the location (\$location) defines the category of the object (page/component) as well. This implies further that it the value of the parameter \$template must provide a valid page or component template.

#### Example:

\$result = createobject ("%publication%", "%page%/%publication%/", "index", "page\_main",
"Miller");

#### Input parameters:

\$site Name of the publication

\$location absolute path (location of the object)

\$object Name oft he new object (page or component)

\$template Name of the page or component template (name of the template

or template file name)

\$user User name

#### global input parameters:

The following global input parameters need to be passed to the function:

\$lang Language setting or language shortcut, e.g. "en", "de"

#### Output:

Array \$result holds the following information:

\$result[result] True/False (result of the action)
\$result[add\_onload] JavaScript code for the onLoad event

\$result[message] Message regarding the result of the action or error message

\$result[object] File name of the page or component \$result[objectname] Name of the page or component \$result[objecttype] File-type or file extension of the file

# 3.5.5 deleteobject

#### Syntax:

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

# **Description:**

Removes an existing page, file or component.

Example:

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

Input parameters:

\$site Name of the publication

\$location absolute path (location of the Object)

\$object Name of the object

\$user User name

#### global input parameters:

The following global input parameters need to be passed to the function:

\$lang Language setting or language shortcut, e.g. "en", "de"

**Output:** 

Array \$result holds the following information:

\$result[result] True/False (result of the action)
\$result[add\_onload] JavaScript code for the onLoad event

\$result[message] Message regarding the result of the action or error message

\$result[publication] Name of the publication where the object is located

\$result[location] absolute path (location of the Object)
\$result[object] File name of the page, file or component
\$result[objectname] Name of the page, file or component
\$result[objecttype] File-type or file extension of the file

# 3.5.6 renameobject

#### Syntax:

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

# **Description:**

Renames an existing page, file or component.

#### Example:

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

#### Input parameters:

\$site Name of the publication

\$location absolute path (location of the Object)

\$object old name of the object

\$objectnew new name of the object (without file extension)

\$user User name

# global input parameters:

The following global input parameters need to be passed to the function:

\$lang Language setting or language shortcut, e.g. "en", "de"

#### Output:

Array \$result holds the following information:

\$result[result] True/False (result of the action)
\$result[add\_onload] JavaScript code for the onLoad event

\$result[message] Message regarding the result of the action or error message

\$result[publication] Name of the publication where the object is located

\$result[location] absolute path (location of the Object)
\$result[object] File name of the page, file or component
\$result[objectname] Name of the page, file or component
\$result[objecttype] File-type or file extension of the file

# 3.5.7 cutobject

#### Syntax:

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

#### **Description:**

Cut an existing page, file or component.

Example:

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

#### Input parameters:

\$site Name of the publication

\$location absolute path (location of the Object)

\$object Name of the object

\$user User name

#### global input parameters:

The following global input parameters need to be passed to the function:

\$lang Language setting or language shortcut, e.g. "en", "de"

#### Output:

Array \$result holds the following information:

\$result[result] True/False (result of the action)
\$result[add\_onload] JavaScript code for the onLoad event

\$result[message] Message regarding the result of the action or error message

\$result[object] File Name of the page, file or component \$result[objectname] Name of the page, file or component \$result[objecttype] File-type or file extension of the file

\$result[clipboard] temporary entry in the clipboard (can be passed as global

variable \$clipboard to the function pasteobject, so reading the

temporary file is not necessary)

# 3.5.8 copyobject

#### Syntax:

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

# Description:

Copy an existing page, file or component.

Example:

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

Input parameters:

\$site Name of the publication

\$location absolute path (location of the Object)

\$object Name of the object

\$user User name

#### global input parameters:

The following global input parameters need to be passed to the function:

\$lang Language setting or language shortcut, e.g. "en", "de"

Output:

Array \$result holds the following information:

\$result[result] True/False (result of the action)
\$result[add\_onload] JavaScript code for the onLoad event

\$result[message] Message regarding the result of the action or error message

\$result[object] File name of the page, file or component \$result[objectname] Name of the page, file or component \$result[objecttype] File-type or file extension of the file

\$result[clipboard] temporary entry in the clipboard (can be passed as global

variable \$clipboard to the function pasteobject, so reading the

temporary file is not necessary)

# 3.5.9 copyconnectedobject

#### Syntax:

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

# Description:

Connected copy an existing page, file or component sharing the same content container.

Example:

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

#### Input parameters:

\$site Name of the publication

\$location absolute path (location of the Object)

\$object Name of the object

\$user User name

### global input parameters:

The following global input parameters need to be passed to the function:

\$lang Language setting or language shortcut, e.g. "en", "de"

# Output:

Array \$result holds the following information:

\$result[result] True/False (result of the action)
\$result[add\_onload] JavaScript code for the onLoad event

\$result[message] Message regarding the result of the action or error message

\$result[object] File name of the page, file or component \$result[objectname] Name of the page, file or component \$result[objecttype] File-type or file extension of the file

\$result[clipboard] temporary entry in the clipboard (can be passed as global

variable \$clipboard to the function pasteobject, so reading the

temporary file is not necessary)

# 3.5.10 pasteobject

#### Syntax:

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

# **Description:**

Paste Einfügen an existing page, file or component.

Example:

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

Input parameters:

\$site Name of the publication

\$location absolute path (location of the Object)

\$user User name

\$clipboard temporary entry in the clipboard (can be passed as global

variable \$clipboard to the function pasteobject, so reading the

temporary file is not necessary)

#### global input parameters:

The following global input parameters need to be passed to the function:

\$lang Language setting or language shortcut, e.g. "en", "de"

**Output:** 

Array \$\text{ Array \$\text{ following information:}}

\$result[result] True/False (result of the action)
\$result[add\_onload] JavaScript code for the onLoad event

\$result[message] Message regarding the result of the action or error message

\$result[publication] Name of the publication where the object is located

\$result[location] absolute path (location of the Object)
\$result[object] File name of the page, file or component
\$result[objectname] Name of the page, file or component
\$result[objecttype] File-type or file extension of the file

# 3.5.11 lockobject

#### Syntax:

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

#### **Description:**

Locking of one or more existing pages or components based on the same content containers for the exclusive use of a user.

# Example:

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

#### Input parameters:

\$site Name of the publication

\$location absolute path (location of the Object)

\$object Name of the object

\$user User name

#### global input parameters:

The following global input parameters need to be passed to the function:

\$lang Language setting or language shortcut, e.g. "en", "de"

# Output:

Array \$result holds the following information:

\$result[result] True/False (result of the action)
\$result[add\_onload] JavaScript code for the onLoad event

\$result[message] Message regarding the result of the action or error message

\$result[object] File name of the page, file or component \$result[objectname] Name of the page, file or component \$result[objecttype] File-type or file extension of the file

# 3.5.12 unlockobject

#### Syntax:

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

#### **Description:**

Unlocking of one or more existing pages or components based on the same content containers for the exclusive use of a user.

#### Example:

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

#### Input parameters:

\$site Name of the publication

\$location absolute path (location of the Object)

\$object Name of the object

\$user User name

### global input parameters:

The following global input parameters need to be passed to the function:

\$lang Language setting or language shortcut, e.g. "en", "de"

# Output:

Array \$result holds the following information:

\$result[result] True/False (result of the action)
\$result[add\_onload] JavaScript code for the onLoad event

\$result[message] Message regarding the result of the action or error message

\$result[object] File name of the page, file or component \$result[objectname] Name of the page, file or component \$result[objecttype] File-type or file extension of the file

# 3.5.13 publishobject

#### Syntax:

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

# **Description:**

Publishing a page or component. All connected copies of the object and its content container will be published as well. If a workflow is in use and does not permit the publishing, the object will not be published.

#### Example:

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

# Input parameters:

\$site Name of the publication

\$location absolute path (location of the Object)

\$object Name of the object

\$user User name

# global input parameters:

The following global input parameters need to be passed to the function:

\$lang Language setting or language shortcut, e.g. "en", "de"

# Output:

Array \$result holds the following information:

\$result[result] True/False (result of the action)
\$result[add\_onload] JavaScript code for the onLoad event

\$result[message] Message regarding the result of the action or error message

# 3.5.14 unpublishobject

#### Syntax:

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

## **Description:**

Unpublishing a page or component. Link and task management will be executed automatically. All connected copies of the object and its content containers will be unpublished as well.

#### Example:

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

### Input parameters:

\$site Name of the publication

\$location absolute path (location of the Object)

\$object Name of the object

\$user User name

### global input parameters:

The following global input parameters need to be passed to the function:

\$lang Language setting or language shortcut, e.g. "en", "de"

#### **Output:**

Array \$result holds the following information:

\$result[result] True/False (result of the action)
\$result[add\_onload] JavaScript code for the onLoad event

\$result[message] Message regarding the result of the action or error message

# 3.5.15 getlinkedobject

#### Syntax:

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

#### **Description:**

This function extracts all objects that have a reference to the given object. This may be page or component links. If the object is a page, then all objects which have a page link to the object will be determined. If the object is a component, all objects which have a component link to the object will be determined.

#### Example:

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

#### Input parameters:

\$site Name of the publication

\$location absolute path (location of the Object)

\$object Name of the object

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

**Output:** 

Array \$result holds the following information:

\$result False (action was not successful)

\$result[publication] Name of the publication where the object is located

\$result[location] absolute path (location of the Object)

\$result[object] Name of the object

\$result[category] Object category [page, comp]

### 3.5.16 getconnectedobject

### Syntax:

getconnectedobject (\$site, \$container)

# **Description:**

This function determines all objects that are based on the same content container. The name of the content container of an object can be by extracted by the function "getfilename".

#### Example:

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

# Input parameters:

\$site Name of the publication \$container Name of the content container

**Output:** 

Array \$result holds the following information:

\$result False (action was not successful)

\$result[publication] Name of the publication where the object is located

\$result[location] absolute path (location of the Object)

\$result[object] Name of the object

\$result[category] Object category [page, comp]

# 3.5.17 getobjectcontainer

# Syntax:

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

#### **Description:**

This function loads the content containers (XML string) of a particular object. The object can be a page, file, component or folder.

The desired content can be extracted from the XML-based container using the function "getcontent" or "selectcontent".

#### Example:

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

### Input parameters:

\$site Name of the publication

\$location absolute path (location of the Object)

\$object Name of the object

\$user User name

**Output:** 

XML-String Content of the content container

False An error occured

### 3.6 Edit content

The programmatic editing of an object contents is shown by the following example. Note the function settext, which is used to manipulate text contents of an object, for more details see the Set API function library.

```
// load object file information
$objectinfo = getobjectinfo ("%publication%", "%page%/%publication%/Home/",
"index.php", "demouser", $container_version="");
// load content container (work status)
$contentdata = loadcontainer ($objectinfo['container_id'], "work", "demouser");
// set a new text
$text = array();
$type = array();
$textuser = array();
$text['Title'] = "My new title";
type['Title'] = "u";
$textuser['Title'] = "demouser";
$text['Description'] = "My new description":
$type['Description'] = "f";
$textuser['Description'] = "demouser";
$containerdata = settext ("%publication%", $contentdata, $objectinfo['container'], $text,
$type, "no", $textuser, "demouser");
// save working xml content container file
if (!empty ($containerdata)) $result = savecontainer ($objectinfo['container_id'], "work",
$containerdata, "demouser");
```

# 3.7 File operation library

The following functions for file operations should never be used to load or save objects (pages, components or files).

However you can use them to load and save XML content container, if you intend to develop extensions or applications.

#### 3.7.1 loadfile

#### Syntax:

loadfile (\$abs\_path, \$filename)

#### **Description:**

This function loads the content of a file. The absolute path and the filename itself must be provided as input parameters. The function waits usually up to 3 seconds to load locked files. If the user parameter \$user is set, the function can also read locked files of the given user.

### Example:

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

#### Input parameters:

\$abs\_path absolute path of the file, %page% and %comp% can be used as the root

elements of the path

\$filename file name

### Output:

File content The function was executed successfully and returns the content of the file

False An error occured

#### 3.7.2 savefile

#### Syntax:

savefile (\$abs\_path, \$filename, \$filedata)

#### **Description:**

This function saved content in files. The absolute path of the file name, and the content that will be written to the file needs to be passed as parameters. If the file is locked, it will not be saved and False will be returned.

# Example:

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

#### Input parameters:

\$abs\_path absolute path of the file, %page% and %comp% can be used as the root

elements of the path

\$filename file name

\$filedata Content that will be saved in the file

### Output:

True The function was executed successfully

#### 3.7.3 loadlockfile

#### Syntax:

loadlockfile (\$user, \$abs\_path, \$filename)

### Description:

This function allows to load the content of a file like the function "loadfile", but it is also triggers a locking mechanism for the file.

The function should only be used when the data will be saved again using the function "savelockfile". This ensures that no other write access by other users can take place. The user, the absolute path and the filename itself must be passed as a parameter to load and lock the file. To save and unlock the file the function "savelockfile" must be used.

#### Example:

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

### Input parameters:

\$user User name of the user who locked the file

\$abs\_path absolute path of the file, %page% and %comp% can be used as the root

elements of the path

\$filename file name

#### Output:

File content The function was executed successfully und liefert den Inhalt der Datei

False An error occured

#### 3.7.4 savelockfile

# Syntax:

savelockfile (\$user, \$abs\_path, \$filename, \$filedata)

# **Description:**

The function "savefile" saved data and unlocks previously opened files using "loadlockfile". For this purpose, the user, the absolute path, the file name, and the content that needs to be written to the file must be passed as parameters.

#### Example:

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

#### Input parameters:

\$user User name of the user who locked the file

\$abs\_path absolute path of the file, %page% and %comp% can be used as the root

elements of the path

\$filename file name

\$filedata Content that will be saved in the file

#### Output:

True The function was executed successfully

#### 3.7.5 lockfile

#### Syntax:

lockfile (\$user, \$abs\_path, \$filename)

#### **Description:**

The function "lockfile" locks a file for a specific user, so its available for the exclusive use. For this purpose, the user, the absolute path and the file name must be passed as a parameters.

#### Example:

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

#### Input parameters:

\$user User name of the user who locked the file

\$abs\_path absolute path of the file, %page% and %comp% can be used as the root

elements of the path

\$filename file name

# Output:

True The function was executed successfully

False An error occured

#### 3.7.6 unlockfile

#### Syntax:

unlockfile (\$user, \$abs\_path, \$filename)

## **Description:**

The function "unlockfile" unlocks files that have been previously locked by "lockfile" or opened by "loadlockfile". For this purpose, the user, the absolute path and the file name must be passed as a parameters.

### Example:

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

#### Input parameters:

\$user User name of the user who locked the file

\$abs\_path absolute path of the file, %page% and %comp% can be used as the root

elements of the path

\$filename file name

# Output:

True The function was executed successfully

#### 3.7.7 deletefile

#### Syntax:

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

#### **Description:**

With "deletefile" files and (empty) folders can be deleted. The absolute path, the file or directory name, and a parameter "recursive", which is either (0) or (1), need to be passed. If recursive is set to 1 the entire contents of the directory will be processed, including subdirectories and their files, using the value 0 only the file or directory (if empty) will be removed.

#### Example:

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

#### Input parameters:

\$abs\_path absolute path of the file, %page% and %comp% can be used as the root

elements of the path

\$file file name

\$recursive 0 or 1, if subdirectories should removed recursively as well

#### Output:

True The function was executed successfully

False An error occured

# 3.7.8 appendfile

#### Syntax:

append (\$abs\_path, \$filename, \$filedata)

#### **Description:**

With "appendfile" content can be added to a file. The function does not overwrite existing data of a file, it appends the data at the file end. For this the absolute path, the file name, and the content that needs to be written to the file must be passed as parameters.

#### Example:

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

#### Input parameters:

\$abs\_path absolute path of the file, %page% and %comp% can be used as the root

elements of the path

\$filename file name

\$filedata Content that will be appended to the file

# Output:

True The function was executed successfully

# 3.8 XML library

The following functions allow yout to read and write content from XML content container. You can optionally query the contents of the container with other technologies that can deal with XML. However, the Edit Content library offers a very simple and performant way of doing this.

# 3.8.1 setxmlparameter

# Syntax:

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

#### Description:

Set a specific value of the XML declaration (1.row).

#### Example

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

#### Input parameters:

\$xmldata XML string that should be manipulated \$parameter Name of the tag that should be manipulated

\$value Value saved in the tag

#### Output:

XML-String Return of the manipulated XML string

False An error occured

#### 3.8.2 getcontent

### Syntax:

getcontent (\$xmldata, \$tag)

# **Description:**

Retrieves the XML content from the content container that is located inside the tags \$tag. An array containing all content or childs found will be returned.

#### Example:

// Get all text-childs from the content container
\$text\_array = getcontent (\$xmldata, "<text>");

// Show all text-childs

foreach (\$text\_array as \$text) echo \$text;

#### Input parameters:

\$xmldata XML string holding the content

\$tag Name of the tag holding the information or child nodes

#### Output:

Array Array holding all found values, the first value can be accessed using the first

array element (Array[0])

# 3.8.3 getxmlcontent

#### Syntax:

getxmlcontent (\$xmldata, \$tag)

#### **Description:**

Retrieves the XML content from a content container that is located inside the tags \$tag and leaves in contrast to the function "getcontent" the XML tags in the return value (array). An entire node (well-formed) will therefore be returned.

An array containing all content and childs found will be returned and can be stored and used in a variable of type array.

#### Example:

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

#### Input parameters:

\$xmldata XML string holding the content

\$tag Name of the tag holding the information or child nodes

#### Output:

Array Array holding all found values, the first value can be accessed using the first

array element (Array[0])

False An error occured

#### 3.8.4 selectcontent

#### Syntax:

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

#### **Description:**

Retrieves the XML content defined by the tag \$parenttag from the content container, where the childtag \$childtag has a certain value \$value.

An array with all items found will be returned and can be stored and used in a variable of type array.

#### Example:

Extract of a content container:

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

// Get all text-childs with id=summary
$text_array = selectcontent ($xmldata, "<text>", "<text_id>", "summary");

// Extract the summary from the found content
foreach ($text_array as $text)
{
    $summary = getcontent ($text, "<textcontent>");
}
```

### Input parameters:

\$xmldata XML string holding the content

\$parenttag Name of the tag holding the information or child nodes

\$childtag optional: XML tag that encloses the information that must be of a certain value optional: Value of the condition, the wildcard character \* can be used at the beginning and/or end of the term and is a wildcard for any further characters.

#### Output:

Array Array holding all found values, the first value can be accessed using the first

array element (Array[0])

False An error occured

#### 3.8.5 selectxmlcontent

#### Syntax:

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

#### **Description:**

Retrieves the XML content defined by the tag \$parenttag from the content container, where the childtag \$childtag has a certain value \$value. In contrast to the function "getcontent" the parent tags are included in the return value (array).

An array with all items found will be returned and can be stored and used in a variable of type array.

#### Example:

```
Extract of a content container:
```

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

</text>

// Get all text-childs with id=summary

$text_array = selectxmlcontent ($xmldata, "<text>", "<text_id>", "summary");

// Extract the summary from the found content

foreach ($text_array as $text)

{
    $summary = getcontent ($text, "<textcontent>");
}
```

### Input parameters:

\$xmldata XML string holding the content

\$parenttag Name of the tag holding the information or child nodes

\$childtag optional: XML tag that encloses the information that must be of a certain value optional: Value of the condition, the wildcard character \* can be used at the beginning and/or end of the term and is a wildcard for any further characters.

#### Output:

Array Array holding all found values, the first value can be accessed using the first

array element (Array[0])

### 3.8.6 deletecontent

#### Syntax:

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

### Description:

Removes the entire XML content defined by the tag \$tagname. For the selection of a certain child the appropriate XML childtag \$condtag and the enclosed information as \$condvalue as condition can be passed.

#### Example:

Extract of a content container:

```
<text>
  <text-id>condition</text_id>
  <textuser>editor1</textuser>
  <textcontent>This is my summary!</textcontent>
</text>
.....
```

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

### Input parameters:

\$xmldata XML string holding the content

\$parenttag Name of the tag holding the information or child nodes that should

be removed

\$condtag optional: XML tag that encloses the information that must be of a certain

value

\$condvalue optional: Value of the condition

Output:

XML-String Return of the manipulated XML string

#### 3.8.7 setcontent

#### Syntax:

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

#### **Description:**

An XML string is passed and within a certain parent node (\$parenttagname) a certain parameter (\$condtag) must exists and must have a certain value (\$condvalue). If the condition is satisfied, the value of the parameter \$tagname will be replaced by the new value \$contentnew.

#### Example:

```
Extract of a content container:
```

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

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

#### Input parameters:

\$xmldata XML string holding the content \$parenttagname optional: XML parent tag

\$tagname optional: XML child tag, that value should be replaced (if the condition

is met)

\$contentnew new value for the XML child tag \$tagmame

\$condtag optional: XML tag that encloses the information that must be of a certain

value

\$condvalue optional: Value for the condition

**Output:** 

XML-String Return of the manipulated XML string

# 3.8.8 updatecontent

#### Syntax:

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

## **Description:**

All XML strings \$xmlnode will be replaced by a new XML string \$xmlnodenew in \$xmldata. This method is faster than "setcontent" when the updated XML node has already been extracted from the container.

#### Example:

Extract of a content container:

```
<text>
  <text_id>condition</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 parameters:

\$xmldata XML string holding the content

\$xmlnode XML string to be replaced (node or substring of \$xmldata)

\$xmInodenew optional: new XML string, if empty, the existing XML string will be

removed.

#### **Output:**

XML-String Return of the manipulated XML string

#### 3.8.9 insertcontent

#### Syntax:

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

#### **Description:**

Inserts an XML string (child node) before the end tag of the given XML parent tag. The modified XML string will be returned.

#### Example:

```
Extract of a content container:
```

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

# Input parameters:

\$xmldata XML string holding the content \$insertxmldata XML string that will be inserted

\$tagname optional: Include xml string before the end tag of the given

XML parent tag

#### Output:

XML-String Return of the manipulated XML string

#### 3.8.10 addcontent

#### Syntax:

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

#### **Description:**

Within a parent node a child node will be added, provided that a certain value in the overlying grandparent node meets the condition. In the child node a value can be set as well. The modified XML string will be returned.

```
Example:
Extract of a 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>
----- The new child node will be inserted here ------
  <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",
```

#### Input parameters:

\$xmldata XML string holding the content \$sub\_xmldata XML string that will be inserted

"<articletextlist>", "<text\_id>", "art1:longtext");

\$grandtagname XML child tag where the \$sub\_xmldata should be embedded optional: XML tag that encloses the information that must be

of a certain value

\$condvalue optional: Value for the condition

\$parenttagname optional: XML child tag, where \$sub\_xmldata should be

embedded

\$tagname optional: Child tag of the embedded XML string where a value

will be set

\$contentnew optional: Value for the \$tagname

Output:

XML-String Return of the manipulated XML string

# 3.9 Meta Data Generator library

This function library allows you to automatically create keyword lists and a description from a given content. This can be used to automatically generate and fill in metadata for pages. The meta data from multimedia files can be extracted and stored in the container of an object.

# 3.9.1 getmetakeywords

#### Syntax:

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

#### Description:

The function requires the text content to be passed as input. All keywords are determined from the text and returned as a keyword list.

#### Example:

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

#### Input parameters:

\$text Content als String

\$language optional: Language [en, de], default is "en" optional: Character set, default is "UTF-8"

#### Output:

Keywords Comma seperated list of keywords

False An error occured

### 3.9.2 getmetadescription

### Syntax:

getdescription (\$text, \$charset)

#### **Description:**

The function requires the text content to be passed as input. A brief description from the given text will be extracted and returned.

# Example:

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

#### Input parameters:

\$text Content as string

\$charset optional: Character set, default is "UTF-8"

#### Output:

Keywords Short description of the content

# 3.10 Notifications library

This function library sends automated messages to a user based on limits of a certain value of a particular field.

The user receives a pre-formatted message with information (links) to all objects that are within the search area (date upper and lower limit).

### 3.10.1 licensenotification

#### Syntax:

licensenotification (\$site, \$cat, \$folderpath, \$text\_id, \$date\_begin, \$date\_end, \$user)

#### **Description:**

The function returns all objects due to the specified search range (location and date limits) and sends an e-mail to a specific user with links to all the affected objects.

#### Example:

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

```
// send mail to Miller
```

\$result = licensenotification ("%publication%", "%comp%/%publication%/images/", "comp",
"valid\_date", "2012-09-01", "2012-09-30", "Miller");

#### Input parameters:

\$site Name of the publication \$cat Object category [page, comp]

\$folderpath Location for the defintion of the search area \$text\_id Text ID of the XML node that need to be analyzed

\$date\_begin Start date for the seach (YYYY-MM-DD) \$date\_end End date for the seach (YYYY-MM-DD)

\$user User name

#### Output:

True Mail wurdwas send successfully

# 4 Components and applications

If applications are integrated into components and variables need to be passed from a page to a component, you need to pay attention to the following:

The components must be integrated via the file system (not via HTTP).

All variables to be passed to the component need to be defined in the component as global.

#### Example:

A page passes a variable to a component.

# Code example of a page:

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

The code of the component must be as followed:

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

In the example, the variable \$test and its value "This is just a test!" will be passed to the component and will be displayed in the presentation of the component.

# 5 Database Connectivity

The Database Connectivity of the hyper Content Management Server allows the connection of different databases for the storage and retrieval of content. Relational databases are widely used as an external content repository.

For this purpose, a corresponding hyperCMS tag for the Database Connectivity need to be present in each template, which then refers to a DB-Connect file.

In this file functions are stored that hyperCMS will call, provided that the template points to the function file.

The contents are read from the database and will be displayed to the editor. If the editor modifies the content, it will be written to the database again. For read and write access different databases can be accessed as well. The functions in the DB Connect File offer only the shell or standardized interface to hyperCMS that needs to be filled by the programmer.

The subject of database integration is complex and need to be treated individual, since existing databases and their information must be integrated. hyperCMS does not provide any ER model or commits itself to specific database vendors. In general it can be said that all the possibilities of PHP can be exploited in order to connect to various data sources.

Besides the necessary parameters for queries to relational databases, the entire content conatiner is passed as an XML string. This allows documents or content from the content repository to be stored as a node in XML databases as well.

You therfore decide from which sources you read and save data. With PHP you have a powerful language that gives you access to all major databases.

More information regarding the functions of PHP can be found here: http://www.php.net

# 5.1 Creating a Database Connectivity

If you want to create a Database Connectivity, you need to make a copy of the the file "db\_connect\_default.php". This file can be found in the root directory for storing the management data under the following path: /data /db\_connect/
The copy of the file should be named according to the database you want to connect with.

Open the file and gain insight into the functions. In the source code you will also find a description of the functions and parameters passed as well as the output parameters.

The following example will show a read access to a MySQL database in order to extract a text content. We assume that in a table "TextContent" the content will be presented by the primary key "container\_id" and "text\_id" as well as the text content "text" itself and the text-type "type". The user and the article ID is not stored separately, this is also not necessary for the uniqueness of the content, because the ID of the content container as well as the ID of the element already provides the primary key.

```
// ============== 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

The hyper Content & Digital Asset Management Server provides an event system that allows automated execution of actions when events in the system will be executed. This can be used, for example, to automate manual processes.

Events are usually started by the user by selecting an action, e.g. publishing a page. If he corresponding event is enabled, the event "onpublishobject" will be called after successful execution of the publication process of the page. The functions defined therein will therfore be executed.

The events of the event system can be defined in the "hypercms\_eventsys.inc.php" file. This file is located in the internal repository in the folder "data/event system". In this file there are also other important instructions that must be followed during the execution of events.

The event system is valid within the management system for all publications. The system is part of the hyperCMS API and is thus performed on each invocation of the API functions.

Events can be activated and deactivated, so that the use of its defined events can be easily controlled in the "hypercms\_eventsys.inc.php" file.

There is a distinction between PRE and POST events. The PRE event will be fired before the actual execution of the called action, while the POST event is called after the successful execution of the action.

#### Example:

When publishing an object, the page "index.php" located at the same position should be automatically published as well, since the page "index.php" is used to generate an overview using a hyperCMS script all objects of the same folder.

# 7 hyperCMS API Function Reference

The documentation of all API functions in the current version is also available on our website hypercms.com. You can view the documentation in its current version directly in the browser. Click on the ?-Icon in the template editor to access the help with all hyperCMS tags and API functions.

# 7.1 Main API Functions

#### 7.1.1 correctnumber

# **Syntax**

correctnumber (\$number)

#### Input parameters

\$number ... formated number [string]

#### global input parameters

\$mgmt\_config

#### **Output**

correct mathematical number / false on error

# 7.1.2 cleancontent

### Description

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

#### Syntax

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

# Input parameters

- \$text ... text [string or array]
- \$charset ... character set [string] (optional)

### global input parameters

• \$mgmt\_config

#### Output

· cleaned text / false on error

# 7.1.3 remove\_utf8\_bom

# Description

Remove UTF-8 BOM sequences

#### **Syntax**

remove\_utf8\_bom (\$text)

### Input parameters

• \$text ... text [string]

### Output

• cleaned text / false on error

# 7.1.4 convert\_utf8

### **Syntax**

convert\_utf8 (\$in)

### Input parameters

• \$in ... expression [string or array]

# Output

• UTF-8 converted expression / original expression on error

### 7.1.5 convertchars

### **Syntax**

convertchars (\$expression, \$charset\_from="UTF-8", \$charset\_to="UTF-8")

#### Input parameters

- \$expression ... expression [string or array]
- \$charset\_from ... input character set [string] (optional)
- \$charset\_to ... output character set [string] (optional)

### global input parameters

• \$mgmt\_config

# Output

• converted expression / original expression on error

# 7.1.6 specialchr

#### **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. #\$...)

#### **Syntax**

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

#### Input parameters

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

# Output

true/false

# 7.1.7 specialchr\_encode

#### **Description**

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

#### **Syntax**

specialchr\_encode (\$expression, \$remove="no")

# Input parameters

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

### global input parameters

- \$user
- \$mgmt\_config
- \$hcms\_lang
- \$lang

### Output

expression without special characters (for file names)

# 7.1.8 specialchr\_decode

# Description

This is the decode function for function specialchr\_encode

#### **Syntax**

specialchr\_decode (\$expression)

#### Input parameters

• \$expression ... expression [string]

### global input parameters

- \$user
- \$mgmt\_config
- \$hcms\_lang
- \$lang

### Output

• expression with special characters (for file names) / false

#### 7.1.9 convertdate

#### **Description**

This function converts a date to a different time zone and format.

# **Syntax**

convertdate (\$date, \$timezone1, \$dateformat1="Y-m-d H:i:s", \$timezone2, \$dateformat2="Y-m-d H:i:s")

#### Input parameters

- \$date ... date and time [string]
- \$timezone1 ... time zone source [string]
- \$dateformat1 ... source date format [string] (optional)
- \$timezone2 ... time zone target [string]
- \$dateformat2 ... target date format [string] (optional)

### Output

converted date as tring / false

### 7.1.10 offsettime

#### **Description**

This function calculates the offset time from UTC (Coordinated Universal Time).

#### **Syntax**

offsettime ()

# Input parameters

# Output

• offset time in hh:mm from UTC

# 7.1.11 object\_exists

# Description

This function verifies if an object exists already.

### **Syntax**

object\_exists (\$path)

#### Input parameters

• \$path ... path to an object [string]

# global input parameters

- \$user
- \$mgmt\_config
- \$hcms\_lang
- \$lang

### Output

• true / false

### 7.1.12 is\_utf8

### Description

This function is an alternative to mb\_check\_encoding (which requires an extra PHP module). It is not failsave!

# Syntax

is\_utf8 (\$str)

### Input parameters

• \$str ... expression [string]

# Output

• if string is utf-8 encoded true / false otherwise

# 7.1.13 is\_latin1

# Description

This function is an alternative to mb\_check\_encoding (which requires an extra PHP module). It is not failsave!

# **Syntax**

is\_latin1 (\$str)

### Input parameters

• \$str ... expression [string]

# Output

• if string is latin 1 encoded true / false otherwise

# 7.1.14 makestring

#### **Syntax**

makestring (\$array)

### Input parameters

• \$array ... (multidimensional) array

# Output

• string including all array values / false on error

# 7.1.15 splitstring

### **Syntax**

splitstring (\$string)

### Input parameters

• \$string ... string with ";" or ", " as seperator [string]

# Output

array with string splitted into array / false on error

### 7.1.16 is\_folder

# Description

Checks if the provided path to an object is a folder

#### **Syntax**

is\_folder (\$path)

## Input parameters

\$path ... path to any object [string]

### global input parameters

\$mgmt\_config

### Output

true / false

# 7.1.17 is\_emptyfolder

### Description

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

#### **Syntax**

is\_emptyfolder (\$dir)

# Input parameters

• \$dir ... path to folder [string]

### global input parameters

• \$mgmt\_config

# Output

true / false

# 7.1.18 is\_supported

### Description

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

#### **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 [array]
- \$file ... file name or file extension [string]

# Output

• true / false

# 7.1.19 is\_cloudstorage

#### **Description**

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

#### **Syntax**

is\_cloudstorage (\$site="")

### Input parameters

• \$site ... publication name [string] (optional)

# global input parameters

• \$mgmt\_config

# Output

• true / false

# 7.1.20 is\_cloudobject

#### Description

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

# **Syntax**

is\_cloudobject (\$file)

# Input parameters

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

### global input parameters

\$mgmt\_config

# Output

true / false

# 7.1.21 is\_date

#### **Description**

This function determines if a string represents a valid date format

#### **Syntax**

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

# Input parameters

- \$date ... date [string]
- \$format ... date format [string] (optional)

# Output

true / false

# 7.1.22 is\_tempfile

# Description

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

# **Syntax**

is\_tempfile (\$path)

# Input parameters

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

### Output

• if file is a temp file true / false on error

# 7.1.23 is\_keyword

# Description

This function checks if the provided expression can be used as a keyword

# **Syntax**

is\_keyword (\$keyword)

## Input parameters

• \$keyword ... keyword [string]

# Output

• if expression can be used as a keyword true / false on error

# 7.1.24 is\_thumbnail

### Description

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

#### **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 [boolean]

# Output

• if file is a thumbnail file true / false on error

# 7.1.25 is\_config

#### **Description**

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

#### **Syntax**

is\_config (\$media)

# Input parameters

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

# Output

• if file is a config file true / false if not

# 7.1.26 is\_encryptedfile

#### **Description**

This function checks if the provided file is encrypted

#### **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

# 7.1.27 is\_document

### Description

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

### **Syntax**

is\_document (\$file)

### Input parameters

• \$file ... file name or file extension [string]

# global input parameters

- \$mgmt\_config
- \$hcms\_ext

# Output

• true / false

# 7.1.28 is\_image

### Description

This function determines if a certain file is an image

#### **Syntax**

is\_image (\$file)

# Input parameters

• \$file ... file name or file extension [string]

### global input parameters

- \$mgmt\_config
- \$hcms\_ext

# Output

true / false

# 7.1.29 is\_rawimage

### Description

This function determines if a certain file is a raw image

#### **Syntax**

is\_rawimage (\$file)

# Input parameters

• \$file ... file name or file extension [string]

# global input parameters

- \$mgmt\_config
- \$hcms\_ext

# Output

• true / false

# 7.1.30 is\_aiimage

### Description

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

### **Syntax**

is\_aiimage (\$file)

### Input parameters

• \$file ... file name or file extension [string]

### Output

• true / false

# 7.1.31 is\_video

# Description

This function determines if a certain file is a video

### **Syntax**

is\_video (\$file)

# Input parameters

• \$file ... file name or file extension [string]

# global input parameters

- \$mgmt\_config
- \$hcms\_ext

# Output

• true / false

# 7.1.32 is\_rawvideo

### Description

This function determines if a certain file is a RAW video

### **Syntax**

is\_rawvideo (\$file)

# Input parameters

• \$file ... file name or file extension [string]

# global input parameters

- \$mgmt\_config
- \$hcms\_ext

### Output

true / false

# 7.1.33 is\_audio

### Description

This function determines if a certain file is an audio file

# **Syntax**

is\_audio (\$file)

# Input parameters

• \$file ... file name or file extension [string]

# global input parameters

- \$mgmt\_config
- \$hcms\_ext

# Output

• true / false

# 7.1.34 is\_mobilebrowser

# Description

Detects mobile browsers (smartphones and tablets)

#### **Syntax**

is\_mobilebrowser ()

# Input parameters

# global input parameters

- \$user
- \$mgmt\_config

# Output

true / false

# 7.1.35 is\_iOS

# Description

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

# **Syntax**

is\_iOS ()

# Input parameters

### global input parameters

• \$user

• \$mgmt\_config

# Output

true / false

# 7.1.36 is\_activelanguage

# Description

This function determines if a language has been enabled for automatic translation in the publication settings

#### **Syntax**

is\_activelanguage (\$site, \$langcode)

# Input parameters

- \$site ... publication name [string]
- \$langcode ... 2-digits language code [string]

# global input parameters

• \$mgmt\_config

### Output

• true / false

# 7.1.37 copyrecursive

#### **Description**

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

### **Syntax**

copyrecursive (\$src, \$dst)

# Input parameters

- \$src ... source directory [string]
- \$dst ... destination directory [string]

### Output

true/false

# 7.1.38 array\_iunique

### **Description**

This function is the case-insensitive form of PHPs array\_unique function

#### **Syntax**

array\_iunique (\$array)

### Input parameters

• \$array ... array [array]

### Output

• unique array / false

# 7.1.39 in\_array\_substr

#### **Description**

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

## **Syntax**

in\_array\_substr (\$search, \$array)

## Input parameters

- \$search ... search expression [string]
- \$array ... array [array]

## Output

true / false

### 7.1.40 createfilename

#### Description

Creates a valid file name without special characters that does not exceed the maximum file name length

#### **Syntax**

createfilename (\$filename)

## Input parameters

• \$filename ... path to file or directory [string]

- \$mgmt\_config
- \$hcms\_lang
- \$lang

• new filename/false

#### 7.1.41 correctfile

#### **Syntax**

correctfile (\$abs\_path, \$filename, \$user="")

## Input parameters

- \$abs\_path ... path to file or directory [string]
- \$filename ... file or directory name [string]
- \$user ... user name [string]

### global input parameters

- \$mgmt\_config
- \$hcms\_lang
- \$lang

# Output

correct filename/false

# 7.1.42 correctpath

#### **Syntax**

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

# Input parameters

- \$path ... path to folder [string]
- \$slash ... directory seperator [string] (optional)

# Output

correct path/false

# 7.1.43 convertpath

### Description

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

## **Syntax**

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

### Input parameters

- \$site ... publication name [string]
- \$path ... content management path to folder or object [string]
- \$cat ... object category [page, comp]

#### global input parameters

- \$user
- \$mgmt\_config
- \$hcms\_lang
- \$lang

# Output

converted path or URL / false on error

### 7.1.44 convertlink

### Description

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

#### **Syntax**

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

## Input parameters

- \$site ... publication name [string]
- \$path ... publication management path to folder or object [string]
- \$cat ... object category [page, comp]

# global input parameters

- \$user
- \$mgmt\_config
- \$publ\_config
- \$hcms\_lang
- \$lang

### Output

converted path or URL / false on error

# 7.1.45 deconvertpath

### **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 might return false.

### **Syntax**

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

### Input parameters

- \$objectpath ... string including path to folder or object [string]
- \$type ... convert to file system path or URL [file, url] (optional)
- \$specialchr\_transform ... transform special characters using specialchr\_encode [boolean] (optional)

## global input parameters

- \$user
- \$mgmt\_config
- \$hcms\_lang
- \$lang

# Output

deconverted path/false

### 7.1.46 deconvertlink

### 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.

#### **Syntax**

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

#### Input parameters

- \$path ... path to folder or object [string]
- \$type ... convert to file system path or URL [file, url]

- \$user
- \$mgmt\_config
- \$publ\_config
- \$hcms\_lang

\$lang

## Output

converted absolute link without host/false

## 7.1.47 mediapublicaccess

### Description

Is the media file public accessible. Has it been published or has the setting 'publicdownload' in the main configuration been enabled.

This function does not include direct links to the media files (used in websites).

#### **Syntax**

mediapublicaccess (\$mediafile)

### Input parameters

• \$mediafile ... media file name [string]

# global input parameters

• \$mgmt\_config

#### Output

• true / false

## 7.1.48 createviewlink

#### 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.

#### **Syntax**

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

#### Input parameters

- \$site ... publication name [string]
- \$mediafile ... media file name [string]
- \$name ... media name [string] (optional)
- \$force\_reload ... force reload [boolean] (optional)
- \$type ... link type [wrapper, download] (optional)

- \$user
- \$mgmt\_config

• URL for download of the multimedia file / false on error

## 7.1.49 createportallink

#### **Description**

The portal access link is used to provide public access without logon based on a certain user profile and user hash that is connected with the portal.

## **Syntax**

createportallink (\$site, \$portal)

### Input parameters

- \$site ... publication name [string]
- \$portal ... portal name [string]

### global input parameters

\$mgmt\_config

# Output

• URL for download of the multimedia file / false on error

#### 7.1.50 createaccesslink

#### **Description**

Creates an access link to any object based on the permissions of the provided user account.

#### **Syntax**

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

- \$site ... publication name [string]
- \$location ... location [string] (optional)
- \$object ... object [string] (optional)
- \$cat ... category [page, comp] (optional)
- \$object\_id ... object-ID [string] (optional)
- \$login ... user name of the user account to be used for access [string]
- \$type ... link type [al, dl] (optional)
- \$lifetime ... token lifetime in seconds [integer] (optional)
- \$formats ... formats defined in main configuration [JSON-string] (optional)

- \$user
- \$mgmt\_config
- \$hcms\_lang
- \$lang

## Output

URL for access to given object / false on error

## 7.1.51 createobjectaccesslink

## Description

Creates an access link to any object based on the permissions of the user account defined in the publication settings.

### **Syntax**

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

### Input parameters

- \$site ... publication name [string] (optional)
- \$location ... location [string] (optional)
- \$object ... object [string] (optional)
- \$cat ... category [page, comp] (optional)
- \$object\_id ... object ID [string] (optional)
- \$container\_id ... container-ID or repository media file [string] (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

### 7.1.52 createwrapperlink

## 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]

## **Syntax**

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

### Input parameters

- \$site ... publication name [string] (optional)
- \$location ... location [string] (optional)
- \$object ... object [string] (optional)
- \$cat ... category [page, comp] (optional)
- \$object\_id ... object ID [string] (optional)
- \$container\_id ... container-ID or repository media file [string] (optional)
- \$type ... media file extension or type based on main config settings [string] (optional)
- \$mediaconfig ... media configuration based on main config settings [string] (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

#### 7.1.53 createdownloadlink

#### **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]

### **Syntax**

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

## Input parameters

- \$site ... publication name [string] (optional)
- \$location ... location [string] (optional)
- \$object ... object [string] (optional)
- \$cat ... category [page, comp] (optional)
- \$object\_id ... object ID [string] (optional)
- \$container\_id ... container-ID or repository media file [string] (optional)
- \$type ... media file extension or type based on main config settings [string] (optional)
- \$mediaconfig ... media configuration based on main config settings [string] (optional)

- \$user
- \$mgmt\_config
- \$hcms\_lang
- \$lang

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

#### 7.1.54 createmultiaccesslink

#### Description

Creates an access link to any object based on the permissions of the provided user account.

#### **Syntax**

createmultiaccesslink (\$multiobject, \$login, \$type="al", \$lifetime=0, \$formats="")

### Input parameters

- \$multiobject ... multiobject [path1|path2|path3] or [array]
- \$login ... user name of the user account to be used for access [string]
- \$type ... link type [al, dl] (optional)
- \$lifetime ... token lifetime in seconds [integer] (optional)
- \$formats ... formats [string] (optional)

#### global input parameters

- \$user
- \$mgmt\_config
- \$hcms\_lang
- \$lang

# Output

URL for access to the requested objects / false on error

#### 7.1.55 createmultidownloadlink

#### 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

zna...media me

3rd...folder

#### **Syntax**

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

- \$site ... publication name [string]
- \$multiobject ... multiobject using | as seperator [string] or [array] (optional)
- \$media ... media file name [string] (optional)
- \$location ... location [string] (optional)

- \$name ... presentation name [string] (optional)
- \$user ... user name [string]
- \$type ... conversion type example: jpg [string]
- \$mediacfg ... media configuration used for conversion (e.g.: 1024x768px) [string]
- \$linktype ... link type [wrapper, download] (optional)
- \$flatzip ... flat hierarchy means no directories [boolean] (optional)

- \$mgmt\_config
- \$mgmt\_compress
- \$pageaccess
- \$compaccess
- \$hiddenfolder
- \$hcms\_linking
- \$globalpermission
- \$setlocalpermission
- \$hcms\_lang
- \$lang

## Output

• URL for download of the requested objects / false on error

### 7.1.56 cleandomain

### **Description**

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

# **Syntax**

cleandomain (\$path)

### Input parameters

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

#### global input parameters

- \$mgmt\_config
- \$hcms\_lang
- \$lang

# Output

cleanded string / false on error

# 7.1.57 fileversion

## Description

Creates a version file name

#### **Syntax**

fileversion (\$file)

### Input parameters

• \$file ... file name [string]

### Output

• versioned file name [string] / false on error

#### 7.1.58 createversion

#### Description

Creates a new version of a multimedia file and container

#### Syntax

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

### Input parameters

- \$site ... publication name [string]
- \$file ... media file name or container name [string]
- \$user ... user name [string] (optional)

### global input parameters

- \$mgmt\_config
- \$mgmt\_mediaoptions
- \$mgmt\_docoptions
- \$hcms\_ext
- \$user

### Output

• true / false

# 7.1.59 rollbackversion

### Description

Makes an older object version to the current version

### **Syntax**

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

### Input parameters

- \$site ... publication name [string]
- \$location ... location [string]
- \$page ... object name [string]
- \$container\_version ... container version name [string]
- \$user ... user name [string] (optional)

### global input parameters

- \$mgmt\_config
- \$mgmt\_mediaoptions
- \$mgmt\_docoptions
- \$hcms\_ext
- \$hcms\_lang
- \$lang

### Output

result\_array

#### 7.1.60 deleteversion

### **Description**

Removes the version of an object

### **Syntax**

deleteversion (\$site, \$container\_version, \$user="sys")

#### Input parameters

- \$site ... publication name [string]
- \$container\_version ... location [string]
- \$user ... object name [string]

#### global input parameters

• \$mgmt\_config

# Output

true / false

### 7.1.61 deleteversions

### Description

Removes all versions of all objects or templates

### **Syntax**

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

### Input parameters

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

### global input parameters

- \$user
- \$mgmt\_config
- \$hcms\_lang
- \$lang

### Output

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

## 7.1.62 loadfile\_header

### **Description**

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

#### **Syntax**

loadfile\_header (\$abs\_path, \$filename)

### Input parameters

- \$abs\_path ... path to file [string]
- \$filename ... file name [string]

# global input parameters

- \$user
- \$mgmt\_config
- \$hcms\_lang
- \$lang

# Output

file content

# 7.1.63 loadfile\_fast

### **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.

#### **Syntax**

loadfile\_fast (\$abs\_path, \$filename)

## Input parameters

- \$abs\_path ... path to file [string]
- \$filename ... file name [string]

## global input parameters

- \$user
- \$mgmt\_config
- \$hcms\_lang
- \$lang

## Output

• file content

## 7.1.64 loadfile

#### **Description**

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

### **Syntax**

loadfile (\$abs\_path, \$filename)

# Input parameters

- \$abs\_path ... path to file [string]
- \$filename ... file name [string]

## global input parameters

- \$user
- \$mgmt\_config
- \$hcms\_lang
- \$lang

# Output

• file content

#### 7.1.65 loadlockfile

#### **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.

#### Syntax

loadlockfile (\$user, \$abs\_path, \$filename, \$force\_unlock=3)

#### Input parameters

- \$user ... user name [string]
- \$abs\_path ... path to file [string]
- \$filename ... file name [string]
- \$force\_unlock ... force unlock of file after x seconds [integer]

### global input parameters

- \$mgmt\_config
- \$hcms\_lang
- \$lang

#### Output

• file content

### 7.1.66 savefile

#### Description

This function saves content to a file

## **Syntax**

savefile (\$abs\_path, \$filename, \$filedata)

### Input parameters

- \$abs\_path ... path to file [string]
- \$filename ... file name [string]
- \$filedata ... file content [string]

- \$user
- \$mgmt\_config
- \$hcms\_lang
- \$lang

true/false

#### 7.1.67 savelockfile

#### 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.

### **Syntax**

savelockfile (\$user, \$abs\_path, \$filename, \$filedata)

### Input parameters

- \$user ... user name [string]
- \$abs\_path ... path to file [string]
- \$filename ... file name [string]
- \$filedata ... file content [string]

## global input parameters

- \$mgmt\_config
- \$hcms\_lang
- \$lang

#### **Output**

true/false

## 7.1.68 appendfile

#### **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.

#### Syntax

appendfile (\$abs\_path, \$filename, \$filedata)

- \$abs\_path ... path to file [string]
- \$filename ... file name [string]
- \$filedata ... file content [string]

- \$user
- \$mgmt\_config
- \$hcms\_lang
- \$lang

# Output

true/false

#### 7.1.69 lockfile

### Description

This functions lockes a file for a specific user

#### **Syntax**

lockfile (\$user, \$abs\_path, \$filename)

## Input parameters

- \$user ... user name [string]
- \$abs\_path ... path to file [string]
- \$filename ... file name [string]

#### global input parameters

- \$mgmt\_config
- \$hcms\_lang
- \$lang

# Output

true/false

# 7.1.70 unlockfile

# Description

This functions unlockes a file for a specific user

## **Syntax**

unlockfile (\$user, \$abs\_path, \$filename)

- \$user ... user name [string]
- \$abs\_path ... path to file [string]
- \$filename ... file name [string]

- \$mgmt\_config
- \$hcms\_lang
- \$lang

## Output

true/false

#### 7.1.71 deletefile

## Description

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

#### **Syntax**

deletefile (\$abs\_path, \$filename, \$recursive=false)

## Input parameters

- \$abs\_path ... path to file [string]
- \$filename ... file or directory name [string]
- \$recursive ... delete all files in directory recursively including symbolic links [boolean]

#### global input parameters

- \$user
- \$mgmt\_config
- \$hcms\_lang
- \$lang

#### Output

true/false

### 7.1.72 restoremediafile

#### Description

Moves an exported media file back to the media repository.

#### **Syntax**

restoremediafile (\$site, \$mediafile)

- \$site ... publication name [string]
- \$mediafile ... media file name [string]

\$mgmt\_config

### Output

result array

# 7.1.73 preparemediafile

### Description

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

#### **Syntax**

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

#### Input parameters

- \$site ... publication name [string]
- \$medialocation ... media file location [string]
- \$mediafile ... media file name [string]
- \$user ... user name [string] (optional)

### global input parameters

• \$mgmt\_config

# Output

result array / false on error

### 7.1.74 deletemediafiles

### Description

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

#### Syntax

deletemediafiles (\$site, \$mediafile, \$delete\_original=false)

# Input parameters

- \$site ... publication name [string]
- \$mediafile ... mediafile name [string]
- \$delete\_original ... delete original media file [boolean] (optional)

- \$user
- \$mgmt\_config
- \$mgmt\_mediaoptions
- \$mgmt\_docoptions
- \$hcms\_ext
- \$hcms\_lang
- \$lang

true/false

### 7.1.75 avoidfilecollision

### **Description**

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

#### Syntax

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

### Input parameters

- \$data ... data string [string] (optional)
- \$force ... force execution [boolean]

## global input parameters

- \$mgmt\_config
- \$site

#### Output

• true / false on error

# 7.1.76 substr\_in\_array

### Description

Searches for substring in array

#### Syntax

substr\_in\_array (\$search, \$array)

- \$search ... search-string [string]
- \$array ... array [array]

array with found values / false

# 7.1.77 downloadobject

#### **Description**

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

### **Syntax**

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

### Input parameters

- \$location ... location [string]
- \$object ... object name [string]
- \$container ... content container [string]
- \$lang ... language [string] (optional)
- \$user ... user name [string] (optional)

#### global input parameters

- \$mgmt\_config
- \$eventsystem
- \$hcms\_lang
- \$lang

# Output

stream of file content / false on error

#### 7.1.78 downloadfile

#### Description

This functions provides a file via http for view or download

#### **Syntax**

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

# Input parameters

- \$filepath ... path to file [string]
- \$name ... file name to show for download via http [string]
- \$force ... force file wrapper or download or no file headers for WebDAV [download, wrapper, noheader]
- \$user ... user name [string] (optional)

- \$mgmt\_config
- \$eventsystem
- \$hcms\_lang
- \$lang
- \$is\_iphone

stream of file content / false on error

# 7.1.79 loadcontainer

#### **Description**

This functions loads a content container

#### **Syntax**

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

## Input parameters

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

#### global input parameters

- \$mgmt\_config
- \$hcms\_lang
- \$lang

#### Output

• XML content of container / false on error

## 7.1.80 savecontainer

### Description

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

#### **Syntax**

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

- \$container ... container file name or container id (working container will be loaded by default) [string]
- \$type ... container type [published, work, version] (optional)

- \$data ... container content [string]
- \$user ... user name [string]
- \$init ... save container initally [boolean] (optional)

- \$mgmt\_config
- \$hcms\_lang
- \$lang

#### Output

true / false on error

#### 7.1.81 checkworkflow

#### **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.

#### **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

# 7.1.82 inherit\_db\_load

### 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

#### **Syntax**

inherit\_db\_load (\$user)

### Input parameters

• \$user ... %

#### global input parameters

- \$siteaccess
- \$mgmt\_config
- \$hcms\_lang
- \$lang

## Output

• inheritance database [2 dim. array]/false

# 7.1.83 inherit\_db\_read

## Description

This function loads the inheritance database for reading

# **Syntax**

inherit\_db\_read ()

# Input parameters

#### global input parameters

- \$user
- \$mgmt\_config
- \$hcms\_lang
- \$lang

## Output

• inheritance database [2 dim. array]/false

# 7.1.84 inherit\_db\_close

#### Description

Closes and unlocks the inheritance management database

### **Syntax**

inherit\_db\_close (\$user)

# Input parameters

• \$user ... %

## global input parameters

- \$mgmt\_config
- \$hcms\_lang
- \$lang

# Output

• true/false

# 7.1.85 inherit\_db\_save

### Description

This function saves und unlocks the inheritance management database

#### **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

# 7.1.86 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

# 7.1.87 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

# 7.1.88 inherit\_db\_setparent

#### Description

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

#### **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]

# 7.1.89 inherit\_db\_insertparent

### **Syntax**

inherit\_db\_insertparent (\$inherit\_db, \$parent, \$child\_array)

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

• \$child\_array ... childs [array]

# Output

• inherit database [2 dim. array]

# 7.1.90 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]

### 7.1.91 createinstance

#### Description

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

#### Syntax

createinstance (\$instance\_name, \$settings, \$user="sys")

### Input parameters

- \$instance\_name ... instance name [string]
- \$settings ... settings array [array]
- \$user ... user name [string]

# global input parameters

- \$eventsystem
- \$mgmt\_config
- \$hcms\_lang
- \$lang

## Output

result array

### 7.1.92 editinstance

### Description

This function saves the instance configuration in the config file

### **Syntax**

editinstance (\$instance\_name, \$content, \$user="sys")

### Input parameters

- \$instance\_name ... instance name [string]
- \$content ... content [string]
- \$user ... user name [string]

# global input parameters

- \$eventsystem
- \$mgmt\_config
- \$hcms\_lang
- \$lang

## Output

result array

### 7.1.93 deleteinstance

### Description

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

### **Syntax**

deleteinstance (\$instance\_name, \$user="sys")

### Input parameters

- \$instance\_name ... instance name [string]
- \$user ... settings [array]

# global input parameters

- \$eventsystem
- \$mgmt\_config
- \$hcms\_lang
- \$lang

# Output

result array

# 7.1.94 createpublication

### Description

This function creates a new publication with all its files

#### **Syntax**

createpublication (\$site\_name, \$user="sys")

#### Input parameters

- \$site\_name ... publication name [string]
- \$user ... user name [string] (optional)

#### global input parameters

- \$siteaccess
- \$eventsystem
- \$mgmt\_config
- \$hcms\_lang
- \$lang

#### **Output**

result array

### 7.1.95 editpublication

```
Description
This function saves all settings of a publication. It is a good advice to load the settings of a
publication and manipulate the values in order to provide all settings as input.
Settings (keys for settings array):
inherit_comp ... inherit components [boolean]
inherit_tpl ... inherit templates [boolean]
youtube_token ... Youtube token [string]
registration ... enable registration of new users [boolean]
registration_group ... assign new users to the group [string]
registration_notify ... notify the users whenever a new user registered [string]
site_admin0 ... enable publication management [true]
url_path_page ... URL to page root on management server [string]
abs_path_page .... absolute path to page root on management server [string]
exclude_folders ... exlude folders from the system [string]
allow_ip ... only enable access for the provided IP addresses [string]
webdav ... enable WebDAV access [boolean]
webdav_dl ... enable download links for WebDAV [boolean]
webdav_al ... enable access links for WebDAV [boolean]
default_codepage ... deafult charcter set of the publication [string]
sendmail ... enable send mail [boolean]
mailserver ... mailserver name [string]
portalaccesslink ... enable portal access via public link [boolean]
accesslinkuser ... user to be used for access links [string]
watermark_image ... watermark settings for images (-wm
/home/hypercms/public html/logo watermark.png->topleft->10) [string]
watermark_video ... watermark settings for videos (-wm
```

```
/home/hypercms/public_html/logo_watermark.png->topleft->10) [string]
sharesociallink ... enable social media link sharing [boolean]
youtube ... enbale Youtube upload [boolean]
theme ... standard design theme of the publication [string]
translate ... languges that can be used for automated translation (en, fr, de) [string]
ocr .. languges that can be used for OCR (en, de) [string]
storage_limit ... storage limit in MB [integer]
gs_access_json ... Google Cloud service JSON access code [string]
gs_speech2text_langcode ... language code to be used for Google Speech2Text Cloud service
(en-US) [string]
url_publ_page ... URL to page root on publication server [string]
abs_publ_page ... absolute path to page root on publication server [string]
url_publ_rep ... URL to repository root on publication server [string]
abs_publ_rep ... absolute path to repository root on publication server [string]
abs_publ_app ... absolute path to application root on publication server [string]
publ_os ... operating system used (UNIX or WIN) [string]
remoteclient ... URL to remote client on remote webserver [string]
```

#### Syntax

editpublication (\$site\_name, \$setting, \$user="sys")

#### Input parameters

- \$site\_name ... publication name [string]
- \$setting ... publication settings with setting name as key and parameter as value [array]
- \$user ... user name [string] (optional)

### global input parameters

- \$eventsystem
- \$mgmt\_config
- \$hcms\_lang
- \$lang

# Output

result array

# 7.1.96 editpublicationsetting

### Description

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

#### Syntax

editpublicationsetting (\$site\_name, \$setting, \$user="sys")

- \$site\_name ... publication name [string]
- \$setting ... publication settings with setting name as key and setting parameter as value (see publication config file for details) [array]
- \$user ... user name [string] (optional)

- \$eventsystem
- \$mgmt\_config
- \$hcms\_lang
- \$lang

## Output

result array

# 7.1.97 deletepublication

## Description

This function deletes a publication with all its files

#### **Syntax**

deletepublication (\$site\_name, \$user="sys")

## Input parameters

- \$site\_name ... publication name [string]
- \$user ... user name [string] (optional)

# global input parameters

- \$mgmt\_config
- \$eventsystem
- \$hcms\_lang
- \$lang

### Output

result array

# 7.1.98 createpersonalization

### Description

This function creates a new customer personalization tracking or profile

## **Syntax**

createpersonalization (\$site, \$pers\_name, \$cat)

- \$site ... publication name [string]
- \$pers\_name ... personalization profile or tracking name [string]
- \$cat ... category [profile, tracking]

- \$user
- \$eventsystem
- \$mgmt\_config
- \$hcms\_lang
- \$lang

### Output

result array

# 7.1.99 deletepersonalization

#### **Description**

This function deletes a customer personalization tracking or profile

#### **Syntax**

deletepersonalization (\$site, \$pers\_name, \$cat)

#### Input parameters

- \$site ... publication name [string]
- \$pers\_name ... personalization profile or tracking name [string]
- \$cat ... category [profile, tracking]

### global input parameters

- \$user
- \$eventsystem
- \$mgmt\_config
- \$hcms\_lang
- \$lang

### Output

result array

# 7.1.100 loadtemplate

## 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.

### **Syntax**

loadtemplate (\$site, \$template)

### Input parameters

- \$site ... publication name [string]
- \$template ... template file name [string]

### global input parameters

- \$user
- \$mgmt\_config
- \$hcms\_lang
- \$lang

## Output

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

# 7.1.101 createtemplate

#### **Description**

This function creates a new template

#### **Syntax**

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

#### Input parameters

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

# global input parameters

- \$user
- \$eventsystem
- \$mgmt\_config
- \$hcms\_lang
- \$lang

## Output

result array

# 7.1.102 edittemplate

### Description

This function edites a template

### **Syntax**

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

### Input parameters

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

### global input parameters

- \$eventsystem
- \$mgmt\_config
- \$hcms\_lang
- \$lang

### Output

result array

# 7.1.103 deletetemplate

### Description

This function deletes a template

#### **Syntax**

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

## Input parameters

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

- \$user
- \$eventsystem
- \$mgmt\_config
- \$hcms\_lang
- \$lang

result array

### 7.1.104 createportal

#### **Description**

This function creates a new portal template

### **Syntax**

createportal (\$site, \$template)

### Input parameters

- \$site ... publication name [string]
- \$template ... portal template name [string]

### global input parameters

- \$user
- \$eventsystem
- \$mgmt\_config
- \$hcms\_lang
- \$lang

### Output

result array

# 7.1.105 editportal

## Description

This function edites the settings of a portal.

### **Syntax**

editportal (\$site, \$template, \$portaluser, \$design="day", \$primarycolor="", \$global\_files=array(), \$formats="", \$user="sys")

- \$site ... publication name [string]
- \$template ... template name or file name [string]
- \$portaluser ... portal user name [string]
- \$design ... design theme name [string] (optional)
- \$primarycolor ... primary color as hex code [string] (optional)
- \$global\_files ... PHP global FILES variable for file upload/remove [array] (optional)
- \$formats ... download formats [JSON-string] (optional)
- \$user ... user name [string] (optional)

- \$eventsystem
- \$mgmt\_config
- \$hcms\_lang
- \$lang

# Output

result array

### 7.1.106 deleteportal

## Description

This function deletes a portal

#### **Syntax**

deleteportal (\$site, \$template)

## Input parameters

- \$site ... publication name [string]
- \$template ... template file name [string]

# global input parameters

- \$user
- \$eventsystem
- \$mgmt\_config
- \$hcms\_lang
- \$lang

#### Output

result array

### 7.1.107 createuser

### Description

This function creates a new user. Use \*Null\* for publication name to remove access to all publications.

#### **Syntax**

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

### Input parameters

• \$site ... publication name [string] (optional)

- \$login ... user login name [string]
- \$password ... password [string]
- \$confirm\_password ... confirmed password [string]
- \$user ... user name [string] (optional)

- \$eventsystem
- \$mgmt\_config
- \$mgmt\_lang\_shortcut\_default
- \$hcms\_lang
- \$lang

## Output

result array

#### 7.1.108 edituser

#### Description

This function edits a user. Use \*Leave\* as input if a value should not be changed. Use \*Null\* for publication name to remove access to all publications. Use \*Null\* for user group to remove user from all user groups of the publication.

#### **Syntax**

```
edituser ($site="", $login, $old_password="", $password="", $confirm_password="", $superadmin="0", $realname="*Leave*", $language="en", $timezone="*Leave*", $theme="*Leave*", $phone="*Leave*", $signature="*Leave*", $usergroup="*Leave*", $validdatefrom="*Leave*", $validdatefrom="*Leave*", $validdateto="*Leave*", $user="sys")
```

- \$site ... publication name [string]
- \$login ... user login name [string]
- \$old\_password ... new login name [string] (optional)
- \$password ... password [string] (optional)
- \$confirm\_password ... confirmed password [string] (optional)
- \$superadmin ... super administrator [0, 1] (optional)
- \$realname ... real name [string] (optional)
- \$language ... language setting [en, de, ...] (optional)
- \$timezone ... time zone [string] (optional)
- \$theme ... theme name (optional)
- \$email ... email [string] (optional)
- \$phone ... phone [string] (optional)
- \$signature ... signature [string] (optional)
- \$usergroup ... member of usergroup string [group1|group2] (optional)
- \$usersite ... member of publications string [site1|site2] (optional)
- \$validdatefrom ... valid date from [date] (optional)
- \$validdateto ... valid date to [date] (optional)
- \$user ... user name [string] (optional)

- \$eventsystem
- \$login\_cat
- \$group
- \$mgmt\_config
- \$hcms\_lang
- \$lang

#### Output

result array

## 7.1.109 deleteuser

#### Description

This function removes a user

#### **Syntax**

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

#### Input parameters

- \$site ... publication where the user should be removed [\*Null\*] for all publications [string]
- \$login ... login name of the account that will be removed [string]
- \$user ... user name [string]

#### global input parameters

- \$eventsystem
- \$mgmt\_config
- \$hcms\_lang
- \$lang

## Output

result array

# 7.1.110 creategroup

#### Description

This function creates a new user group

#### **Syntax**

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

- \$site ... publication name [string]
- \$groupname ... group name [string]
- \$user ... user name [string]

- \$eventsystem
- \$mgmt\_config
- \$hcms\_lang
- \$lang

## Output

result array

# 7.1.111 editgroup

## Description

This function edits the settings of a user group.

The following permissions can be used:

desktopglobal

desktopsetting

desktoptaskmgmt

desktopcheckedout

desktoptimetravel

userglobal

usercreate

userdelete

useredit

groupglobal

groupcreate

groupdelete

groupedit

siteglobal

sitecreate

sitedelete

siteedit

persglobal

perstrack

perstrackcreate

perstrackdelete

perstrackedit

persprof

persprofcreate

persprofdelete

persprofedit

workflowglobal

workflowproc

workflowproccreate

workflowprocdelete

workflowprocedit

workflowprocfolder

workflowscript

workflowscriptcreate

workflowscriptdelete

workflowscriptedit

templateglobal

tpl

tplcreate

tpldelete

tpledit

tplmedia

tplmediacatcreate

tplmediacatdelete

tplmediacatrename

tplmediaupload tplmediadelete

componentglobal

compupload

compdownload

compsendlink

compfoldercreate

compfolderdelete

compfolderrename

compcreate

compdelete

comprename

comppublish

pageglobal

pagesendlink

pagefoldercreate

pagefolderdelete

pagefolderrename

pagecreate

pagedelete

pagerename

pagepublish

The value of each parameter can be 1 for granted permission or 0 for no permission. If you need further explanation about the permissions, please have a look at the

Administrators Guide.

#### **Syntax**

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

#### Input parameters

- \$site ... publication name [string]
- \$groupname ... group name [string]
- \$pageaccess ... page folder access array [array]
- \$compaccess ... component folder access array [array]
- \$permission ... permissions with permission name as key and value of 1 or 0 [array]
- \$user ... user name [string]

#### global input parameters

- \$eventsystem
- \$mgmt\_config
- \$hcms\_lang
- \$lang

result array

# 7.1.112 deletegroup

## Description

This function removes a user group

### **Syntax**

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

## Input parameters

- \$site ... publication name [string]
- \$groupname ... group name [string]
- \$user ... user name [string]

#### global input parameters

- \$eventsystem
- \$mgmt\_config
- \$hcms\_lang
- \$lang

## Output

result array

# 7.1.113 renamegroupfolder

### Description

This function renames a workgroup folder

## **Syntax**

renamegroupfolder (\$site, \$cat, \$folder\_curr, \$folder\_new, \$user)

## Input parameters

- \$site ... publication name [string]
- \$cat ... category [page, comp]
- \$folder\_curr ... old location [string]
- \$folder\_new ... new location [string]
- \$user ... user name [string]

#### global input parameters

• \$mgmt\_config

- \$hcms\_lang
- \$lang

true / false on error

# 7.1.114 deletegroupfolder

#### **Description**

This function removes a group folder.

#### Syntax

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

## Input parameters

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

### global input parameters

- \$mgmt\_config
- \$hcms\_lang
- \$lang

## Output

• true / false on error

## 7.1.115 renameworkflowfolder

## Description

This function renames a workgroup folder

# **Syntax**

renameworkflowfolder (\$site, \$cat, \$folder\_curr, \$folder\_new, \$user)

- \$site ... publication name [string]
- \$cat ... category [page, comp]
- \$folder\_curr ... old location [string]
- \$folder\_new ... new location [string]
- \$user ... user name [string]

- \$mgmt\_config
- \$hcms\_lang
- \$lang

### Output

true / false on error

#### 7.1.116 deleteworkflowfolder

#### Description

This function removes a workgroup folder

#### Syntax

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

#### Input parameters

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

#### global input parameters

- \$mgmt\_config
- \$hcms\_lang
- \$lang

# Output

• true / false on error

## 7.1.117 createmediacat

# Description

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

#### Svntax

createmediacat (\$site, \$mediacat\_name)

- \$site ... publication name [string]
- \$mediacat\_name ... media category name [string]

- \$user
- \$eventsystem
- \$mgmt\_config
- \$hcms\_lang
- \$lang

# Output

Array with onload JS-code and message

#### 7.1.118 renamemediacat

#### **Description**

Renames a new media category in the template media index file

#### **Syntax**

renamemediacat (\$site, \$mediacat\_name\_curr, \$mediacat\_name)

## Input parameters

- \$site ... publication name [string]
- \$mediacat\_name\_curr ... old media category name [string]
- \$mediacat\_name ... new media category name [string]

## global input parameters

- \$user
- \$eventsystem
- \$mgmt\_config
- \$hcms\_lang
- \$lang

## Output

Array with onload JS-code and message

## 7.1.119 deletemediacat

# Description

Deletes a new media category in the template media index file

### **Syntax**

deletemediacat (\$site, \$mediacat\_name)

- \$site ... publication name [string]
- \$mediacat\_name ... media category name [string]

- \$user
- \$eventsystem
- \$mgmt\_config
- \$hcms\_lang
- \$lang

## Output

Array with onload JS-code and message

# 7.1.120 uploadtomediacat

#### Description

Uploads a media file into a given template media category

#### **Syntax**

uploadtomediacat (\$site, \$mediacat\_name, \$global\_files)

## Input parameters

- \$site ... publication name [string]
- \$mediacat\_name ... media category name [string]
- \$global\_files ... PHP global FILES variable [array]

## global input parameters

- \$user
- \$eventsystem
- \$mgmt\_config
- \$hcms\_lang
- \$lang

## Output

Array with onload JS-code and message

## 7.1.121 deletefrommediacat

#### **Description**

Deletes a media file from the template media category index

#### Syntax

deletefrommediacat (\$site, \$mediafile)

## Input parameters

- \$site ... publication name [string]
- \$mediafile ... media file name [string]

## global input parameters

- \$user
- \$eventsystem
- \$mgmt\_config
- \$hcms\_lang
- \$lang

# Output

• Array with onload JS-code and message

## 7.1.122 createfolder

#### **Description**

This function creates a new folder. The folder name must not match any temp file pattern.

#### **Syntax**

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

#### Input parameters

- \$site ... publication name [string]
- \$location ... location [string]
- \$folder ... folder name [string]
- \$user ... user name [string]

#### global input parameters

- \$eventsystem
- \$mgmt\_config
- \$cat
- \$pageaccess
- \$compaccess
- \$hiddenfolder
- \$hcms\_linking
- \$hcms\_lang
- \$lang

#### Output

## 7.1.123 createfolders

### Description

This function creates all folders recursively

#### **Syntax**

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

## Input parameters

- \$site ... publication name [string]
- \$location ... location [string]
- \$folder ... folder name [string]
- \$user ... user name [string]

## global input parameters

- \$eventsystem
- \$mgmt\_config
- \$cat
- \$pageaccess
- \$compaccess
- \$hiddenfolder
- \$hcms\_linking
- \$hcms\_lang
- \$lang

## Output

result array

# 7.1.124 collectfolders

### Description

Help function to create the collection of folders

#### Syntax

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

#### Input parameters

- \$site ... publication name [string]
- \$location ... location [string]
- \$folder ... folder name [string]

#### global input parameters

- \$user
- \$mgmt\_config
- \$hcms\_lang
- \$lang

· result array / false

## 7.1.125 copyfolders

## Description

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

#### **Syntax**

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

#### Input parameters

- \$site ... publication name [string]
- \$location ... location (source) [string]
- \$locationnew ... new location (destination) [string]
- \$folder ... folder name [string]
- \$user ... user name [string]

#### global input parameters

- \$mgmt\_config
- \$cat
- \$pageaccess
- \$compaccess
- \$hiddenfolder
- \$hcms\_linking
- \$hcms\_lang
- \$lang

#### Output

• result array equal to createfolder

## 7.1.126 deletefolder

#### Description

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

#### **Syntax**

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

- \$site ... publication name [string]
- \$location ... location [string]
- \$folder ... folder name [string]

• \$user ... user name [string]

# global input parameters

- \$eventsystem
- \$mgmt\_config
- \$cat
- \$pageaccess
- \$compaccess
- \$hiddenfolder
- \$hcms\_linking
- \$hcms\_lang
- \$lang

# Output

result array

## 7.1.127 renamefolder

## Description

This function renames a folder

#### **Syntax**

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

## Input parameters

- \$site ... publication name [string]
- \$location ... location [string]
- \$folder ... folder name [string]
- \$foldernew ... new folder name [string]
- \$user ... user name [string]

# global input parameters

- \$eventsystem
- \$mgmt\_config
- \$cat
- \$pageaccess
- \$compaccess
- \$hiddenfolder
- \$hcms\_linking
- \$hcms\_lang
- \$lang

## Output

## 7.1.128 correct container name

#### **Description**

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

## **Syntax**

correctcontainername (\$container\_id)

## Input parameters

• \$container\_id ... container ID [string]

## Output

corrected name / false on error

# 7.1.129 createobject

#### **Description**

This function creates a new page or component.

### **Syntax**

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

## Input parameters

- \$site ... publication name [string]
- \$location ... location [string]
- \$page ... object name without file extension [string]
- \$template ... template name [string]
- \$user ... user name [string]

## global input parameters

- \$eventsystem
- \$mgmt\_config
- \$pageaccess
- \$compaccess
- \$hiddenfolder
- \$hcms\_linking
- \$hcms\_lang
- \$lang

#### Output

## 7.1.130 uploadfile

#### **Description**

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

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

## **Syntax**

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

#### Input parameters

- \$site ... publication name [string]
- \$location ... destination location [string]
- \$cat ... category [page, comp]
- \$global\_files ... uploaded file (array as defined by PHP autoglobale \$\_FILES) [array]
- \$page ... unzip/zip [%, unzip, zip]
- \$unzip ... object name (only for media file update of existing object) [string]
- \$createthumbnail ... create only a new thumbnail from the media file [1, 0] (optional)
- \$imageresize ... resize image [percentage, null] (optional)
- \$imagepercentage ... image resize percentage value [integer] (optional)
- \$user ... user name [string] (optional)
- \$checkduplicates ... check for duplicates [boolean] (optional)
- \$versioning ... versioning of file [boolean] (optional)
- \$zipfilename ... name of zip file [string] (optional)
- \$zipfilecount ... number of files to be compressed [integer] (optional)

#### global input parameters

- \$mgmt\_config
- \$mgmt\_uncompress
- \$mgmt\_compress
- \$mgmt\_imagepreview
- \$mgmt\_mediapreview
- \$mgmt mediaoptions
- \$mgmt\_imageoptions
- \$mgmt\_maxsizepreview
- \$mgmt\_parser
- \$eventsystem
- \$pageaccess
- \$compaccess
- \$hiddenfolder
- \$localpermission
- \$hcms\_lang
- \$lang

#### Output

# 7.1.131 createmediaobject

#### **Description**

This function creates an asset (multimedia object) by reading a given source file. The file name must not match any temp file pattern.

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

#### **Syntax**

createmediaobject (\$site, \$location, \$file, \$path\_source\_file, \$user, \$imagepercentage=0, \$leavefile=false)

#### Input parameters

- \$site ... publication name [string]
- \$location ... destination location [string]
- \$file ... file name [string]
- \$path\_source\_file ... path to source multimedia file (uploaded file in temp directory) [string]
- \$user ... user name [string]
- \$imagepercentage ... resize original image (100%) by percentage [integer] (optional)
- \$leavefile ... leave file in the directory and create a symbolic link to the file [boolean] (optional)

### global input parameters

- \$mgmt\_config
- \$mgmt\_imagepreview
- \$mgmt\_mediapreview
- \$mgmt\_mediaoptions
- \$mgmt\_imageoptions
- \$mgmt\_maxsizepreview
- \$mgmt\_mediametadata
- \$mgmt\_parser
- \$mgmt\_imagepreview
- \$mgmt\_uncompress
- \$hcms\_ext
- \$eventsystem
- \$pageaccess
- \$compaccess
- \$hiddenfolder
- \$hcms\_linking
- \$hcms\_lang
- \$lang

## Output

result array

# 7.1.132 createmediaobjects

## Description

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

The file name must not match any temp file pattern.

#### **Syntax**

createmediaobjects (\$site, \$location\_source, \$location\_destination, \$user)

#### Input parameters

- \$site ... publication name [string]
- \$location\_source ... source location [string]
- \$location\_destination ... destination location [string]
- \$user ... user name [string]

#### global input parameters

- \$mgmt\_config
- \$mgmt\_imageoptions
- \$eventsystem
- \$pageaccess
- \$compaccess
- \$hiddenfolder
- \$hcms\_linking
- \$hcms\_lang
- \$lang

## Output

result array with all objects created / false

# 7.1.133 editmediaobject

#### **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

#### **Syntax**

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

#### Input parameters

- \$site ... publication name [string]
- \$location ... location [string]
- \$page ... object name [string]
- \$format ... format (file extension w/o dot) [string] (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
- \$mgmt\_mediaoptions
- \$mgmt\_imageoptions
- \$mgmt\_maxsizepreview
- \$mgmt\_mediametadata
- \$hcms\_ext
- \$hcms\_lang
- \$lang

- 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

# 7.1.134 manipulateobject

#### Description

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

#### **Syntax**

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

## Input parameters

- \$site ... publication name [string]
- \$location ... location [string]
- \$page ... object name [string]
- \$pagenew ... new object name (exkl. extension except for action "file\_rename") [string]
- \$user ... user name [string]
- \$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

array

# 7.1.135 deletemarkobject

## Description

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

#### **Syntax**

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

## Input parameters

- \$site ... publication name [string]
- \$location ... location [string]
- \$page ... object name [string]
- \$user ... user name [string]

## global input parameters

- \$wf\_token
- \$eventsystem
- \$mgmt\_config
- \$cat
- \$pageaccess
- \$compaccess
- \$hiddenfolder
- \$hcms\_linking
- \$hcms\_lang
- \$lang

#### Output

result array

# 7.1.136 deleteunmarkobject

#### **Description**

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

## **Syntax**

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

- \$site ... publication name [string]
- \$location ... location [string]
- \$page ... object name [string]
- \$user ... user name [string]

- \$wf\_token
- \$eventsystem
- \$mgmt\_config
- \$cat
- \$pageaccess
- \$compaccess
- \$hiddenfolder
- \$hcms\_linking
- \$hcms\_lang
- \$lang

## Output

result array

# 7.1.137 deleteobject

## Description

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

#### Syntax

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

## Input parameters

- \$site ... publication name [string]
- \$location ... location [string]
- \$page ... object name [string]
- \$user ... user name [string]

## global input parameters

- \$eventsystem
- \$mgmt\_config
- \$cat
- \$pageaccess
- \$compaccess
- \$hiddenfolder
- \$hcms\_linking
- \$hcms\_lang
- \$lang

# Output

# 7.1.138 renameobject

#### Description

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

#### **Syntax**

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

## Input parameters

- \$site ... publication name [string]
- \$location ... location [string]
- \$page ... object name [string]
- \$pagenew ... new object name without file extension [string]
- \$user ... user name [string]

#### global input parameters

- \$eventsystem
- \$mgmt\_config
- \$cat
- \$pageaccess
- \$compaccess
- \$hiddenfolder
- \$hcms\_linking
- \$hcms\_lang
- \$lang

#### Output

result array

### 7.1.139 renamefile

# Description

This function renames a file (NOT a page, component or asset) 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 for renameobject.

#### **Syntax**

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

- \$site ... publication name [string]
- \$location ... location [string]
- \$page ... object name [string]
- \$pagenew ... new object including file extension [string]
- \$user ... user name [string]

- \$eventsystem
- \$mgmt\_config
- \$cat
- \$pageaccess
- \$compaccess
- \$hiddenfolder
- \$hcms\_linking
- \$hcms\_lang
- \$lang

## Output

result array

# 7.1.140 cutobject

#### Description

This function cuts a page, component or asset.

#### **Syntax**

cutobject (\$site, \$location, \$page, \$user, \$clipboard\_add=false, \$clipboard\_session=true)

### Input parameters

- \$site ... publication name [string]
- \$location ... location [string]
- \$page ... object name [string]
- \$user ... user name [string]
- \$clipboard\_add ... add to existing clipboard entries [boolean] (optional)
- \$clipboard\_session ... save clipboard in session [boolean] (optional)

## global input parameters

- \$eventsystem
- \$mgmt\_config
- \$cat
- \$hcms\_lang
- \$lang

#### Output

result array

# 7.1.141 copyobject

## Description

This function copies a page, component or asset.

#### **Syntax**

copyobject (\$site, \$location, \$page, \$user, \$clipboard\_add=false, \$clipboard\_session=true)

#### Input parameters

- \$site ... publication name [string]
- \$location ... location [string]
- \$page ... object name [string]
- \$user ... user name [string]
- \$clipboard\_add ... add to existing clipboard entries [boolean] (optional)
- \$clipboard\_session ... save clipboard in session [boolean] (optional)

## global input parameters

- \$eventsystem
- \$mgmt\_config
- \$cat
- \$hcms\_lang
- \$lang

## Output

result array

# 7.1.142 copyconnectedobject

# Description

This function makes a connected copy of a page, component or asset.

#### **Syntax**

copyconnectedobject (\$site, \$location, \$page, \$user, \$clipboard\_add=false, \$clipboard\_session=true)

## Input parameters

- \$site ... publication name [string]
- \$location ... location [string]
- \$page ... object name [string]
- \$user ... user name [string]
- \$clipboard\_add ... add to existing clipboard entries [boolean] (optional)
- \$clipboard\_session ... save clipboard in session [boolean] (optional)

# global input parameters

- \$eventsystem
- \$mgmt\_config
- \$cat
- \$hcms\_lang
- \$lang

result array

# 7.1.143 pasteobject

## Description

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

### **Syntax**

pasteobject (\$site, \$location, \$user, \$clipboard\_array=array())

## Input parameters

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

## global input parameters

- \$eventsystem
- \$mgmt\_config
- \$cat
- \$pageaccess
- \$compaccess
- \$hiddenfolder
- \$hcms\_linking
- \$hcms\_lang
- \$lang

#### Output

result array

# 7.1.144 lockobject

## Description

This function locks an object for a specific user

#### Syntax

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

- \$site ... publication name [string]
- \$location ... location [string]
- \$page ... object name [string]
- \$user ... user name [string]

- \$eventsystem
- \$mgmt\_config
- \$cat
- \$hcms\_lang
- \$lang

## Output

result array

# 7.1.145 unlockobject

## Description

This function unlocks an object of a specific user

## **Syntax**

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

## Input parameters

- \$site ... publication name [string]
- \$location ... location [string]
- \$page ... object name [string]
- \$user ... user name [string]

## global input parameters

- \$eventsystem
- \$mgmt\_config
- \$cat
- \$hcms\_lang
- \$lang

## Output

array

# 7.1.146 publishobject

#### Description

This function publishes a page, component or asset

#### **Syntax**

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

- \$site ... publication name [string]
- \$location ... location [string]
- \$page ... object name (full name incl. extension) [string]
- \$user ... user name [string]

- \$eventsystem
- \$mgmt\_config
- \$cat
- \$ctrlreload
- \$pageaccess
- \$compaccess
- \$hiddenfolder
- \$hcms\_linking
- \$hcms\_lang
- \$lang

#### Output

result array

# 7.1.147 publishlinkedobject

#### 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.

#### **Syntax**

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

#### Input parameters

- \$site ... publication name [string]
- \$location ... location [string]
- \$page ... object name [string]
- \$user ... user name [string]

## global input parameters

- \$eventsystem
- \$mgmt\_config
- \$cat
- \$ctrlreload
- \$hcms\_lang
- \$lang

# Output

array

## 7.1.148 unpublishobject

## Description

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

## **Syntax**

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

#### Input parameters

- \$site ... publication name [string]
- \$location ... location [string]
- \$page ... object name [string]
- \$user ... user name [string]

## global input parameters

- \$eventsystem
- \$mgmt\_config
- \$cat
- \$ctrlreload
- \$pageaccess
- \$compaccess
- \$hiddenfolder
- \$hcms\_linking
- \$hcms\_lang
- \$lang

#### **Output**

result array

# 7.1.149 processobjects

## Description

Publish, unpublish or delete all objects recursively, and send mails stored in the queue. 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!

#### Syntax

processobjects (\$action, \$site, \$location, \$file, \$published\_only=false, \$user)

- \$action ... action [publish, unpublish, delete]
- \$site ... publication name [string]
- \$location ... location [string]

- \$file ... object name or mail ID [string]
- \$published\_only ... only published objects [boolean]
- \$user ... user name [string]

- \$eventsystem
- \$mgmt\_config
- \$pageaccess
- \$compaccess
- \$hiddenfolder
- \$hcms\_linking
- \$hcms\_lang
- \$lang

### Output

true/false

# 7.1.150 collectobjects

## Description

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

#### Svntax

collectobjects (\$root\_id, \$site, \$cat, \$location, \$published\_only=false)

## Input parameters

- \$root\_id ... root ID [string]
- \$site ... publication name [string]
- \$cat ... category [page, comp]
- \$location ... location [string]
- \$published\_only ... collect only published objects [boolean]

## global input parameters

- \$user
- \$pageaccess
- \$compaccess
- \$mgmt\_config
- \$hiddenfolder
- \$hcms\_lang
- \$lang

## Output

· result array / false

# 7.1.151 manipulateallobjects

#### **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.

#### **Syntax**

manipulateallobjects (\$action, \$objectpath\_array, \$method="", \$force="start", \$published\_only=false, \$user, \$tempfile="", \$maxitems=10)

#### 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

## 7.1.152 createqueueentry

#### Description

Creates a new item in the queue

#### **Syntax**

createqueueentry (\$action, \$object, \$date, \$published\_only, \$data="", \$user)

- \$action ... action [publish, unpublish, delete, mail]
- \$object ... object path [string] or object ID [integer]
- \$date ... date and time [YYY-MM-DD HH:MM]
- \$published\_only ... publish only published objects [boolean]
- \$data ... data to be saved in queue [array] (optional)
- \$user ... user name [string]

\$mgmt\_config

## Output

• true / false

# 7.1.153 savemessage

#### Description

Saves the data of a sent e-mail message.

#### **Syntax**

savemessage (\$data, \$type="mail", \$user)

#### Input parameters

- \$data ... data to be saved in queue [array]
- \$type ... message type [mail, chat] (optional)
- \$user ... user name [string]

#### global input parameters

• \$mgmt\_config

#### Output

true / false

#### 7.1.154 remoteclient

#### **Description**

Sends data to remote client via http post

#### Syntax

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

- \$action ... action [save, copy, delete, rename, get]
- \$root ... root [abs\_path\_link, abs\_path\_media, abs\_path\_comp, abs\_path\_page, abs\_path\_rep]
- \$site ... publication name [string]
- \$location ... location [string]
- \$locationnew ... new location [string]
- \$page ... object name [string]
- \$pagenew ... new object name [string]

- \$user
- \$mgmt\_config
- \$hcms\_lang
- \$lang

## Output

• http answer [string] or false

# 7.1.155 HTTP\_Post

## Description

Sends data via http post and returns response

#### **Syntax**

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

#### Input parameters

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

## Output

• http response [string] / false on error

# 7.1.156 HTTP\_Get

#### Description

Sends data via http get and returns response

#### Syntax

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

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

• http response [string] / false on error

# 7.1.157 HTTP\_Proxy

# Description

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

### **Syntax**

HTTP\_Proxy (\$URL, \$enable\_file=false)

## Input parameters

- \$URL ... URL [string]
- \$enable\_file ... enable post of files [boolean] (optional)

#### global input parameters

\$mgmt\_config

## Output

• http response [string] / false on error

#### 7.1.158 loadbalancer

## 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.

#### **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

# 7.1.159 savelog

## Description

Adds new entries to log file.

An error entry must be formed like:
date[YYYY-MM-DD hh: mm]|name of scipt file|error type: "error", "warning" or
"information"|unique error code in script file|error message

#### **Syntax**

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

### Input parameters

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

### global input parameters

- \$user
- \$login
- \$site
- \$eventsystem
- \$mgmt\_config
- \$hcms\_lang
- \$lang

## Output

• true / false on error

# 7.1.160 loadlog

#### Description

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

## **Syntax**

loadlog (\$logfile="event", \$return\_type="array")

### Input parameters

- \$logfile ... name of log file without extension [string] (optional)
- \$return\_type ... return type [string, array] (optional)

#### global input parameters

- \$user
- \$eventsystem
- \$mgmt\_config
- \$hcms\_lang
- \$lang

• true / false on error

# 7.1.161 deletelog

## Description

Deletes a log file.

## **Syntax**

deletelog (\$logname="")

## Input parameters

• \$logname ... log name [string] (optional)

## global input parameters

- \$user
- \$eventsystem
- \$mgmt\_config
- \$hcms\_lang
- \$lang

# Output

result array

# 7.1.162 debuglog

## Description

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

## **Syntax**

debuglog (\$code)

# Input parameters

• \$code ... code to write to debug file [string]

## global input parameters

- \$user
- \$mgmt\_config
- \$hcms\_lang
- \$lang

• true / false

## 7.1.163 notifyusers

#### Description

Notifies all users based on the given event and location

### **Syntax**

notifyusers (\$site, \$location, \$object, \$event, \$user\_from)

#### Input parameters

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

#### global input parameters

- \$user
- \$mgmt\_config
- \$hcms\_lang\_codepage
- \$hcms\_lang
- \$lang

#### Output

• true / false on error

#### 7.1.164 sendlicensenotification

## 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.

#### **Syntax**

sendlicensenotification (\$site, \$cat, \$folderpath, \$text\_id, \$date\_begin, \$date\_end, \$user, \$format="%Y-%m-%d")

- \$site ... publication name [string]
- \$cat ... category [page, comp]
- \$folderpath ... folder path [string]
- \$text\_id ... text ID for text field [string]
- \$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)

- \$eventsystem
- \$mgmt\_config
- \$hcms\_lang\_codepage
- \$hcms\_lang
- \$lang

## Output

true / false on error

## 7.1.165 licensenotification

#### **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.

#### **Syntax**

licensenotification ()

#### Input parameters

#### global input parameters

- \$eventsystem
- \$mgmt\_config
- \$hcms\_lang\_codepage
- \$hcms\_lang
- \$lang

# Output

• true / false on error

## 7.1.166 sendresetpassword

### Description

Send a new password to the users e-mail address.

### **Syntax**

sendresetpassword (\$login, \$link=false, \$instance="")

- \$login ... user name [string]
- \$link ... provide logon link [boolean] (optional)
- \$instance ... instance name [string] (optional)

### global input parameters

- \$eventsystem
- \$mgmt\_config
- \$hcms\_lang
- \$lang

## Output

message as string

## 7.1.167 html\_diff

## 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 <del>, which can easily be styled with CSS.

#### **Syntax**

html\_diff (\$old, \$new)

## Input parameters

- \$old ... old text [string]
- \$new ... new text [string]

#### Output

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

### 7.1.168 createfavorite

#### **Syntax**

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

### Input parameters

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

## global input parameters

\$mgmt\_config

## Output

• true / false

## 7.1.169 deletefavorite

## **Syntax**

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

## Input parameters

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

### global input parameters

• \$mgmt\_config

## Output

• true / false

## 7.1.170 rewrite\_targetURI

#### **Syntax**

rewrite\_targetURI (\$site, \$text\_id, \$uri, \$exclude\_dir\_esc="", \$rewrite\_type="include")

## Input parameters

- \$site ... publication name [string]
- \$text\_id ... text ID array (text-ID as key and URL parameters as value) [string]
- \$uri ... requested URI [string]
- \$exclude\_dir\_esc ... exclude path [array] (optional)
- \$rewrite\_type ... rewrite type [none, forward, include] (optional)

#### global input parameters

- \$mgmt\_config
- \$publ\_config

HTTP header with target URL as location / false on error

## 7.1.171 rewrite\_homepage

#### **Description**

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

### **Syntax**

rewrite\_homepage (\$site, \$rewrite\_type="forward")

### Input parameters

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

### global input parameters

- \$mgmt\_config
- \$publ\_config

## Output

• HTTP header with target URL as location / false on error

## 7.1.172 load\_csv

#### Description

Analyzes the content from the CSV file and detects delimiter and enclosure characters if left empty. On success the data will be returned as array starting with row index of 1.

#### **Syntax**

load\_csv (\$file, \$delimiter=";", \$enclosure=""', \$charset="utf-8")

#### Input parameters

- \$file ... path to CSV file [string]
- \$delimiter ... delimiter [string] (optional)
- \$enclosure ... enclosure [string] (optional)
- \$charset ... character set [string] (optional)

### global input parameters

- \$mgmt\_config
- \$eventsystem

array / false on error

# 7.1.173 sendmessage

## Description

Sends a message via e-mail to a user.

#### **Syntax**

sendmessage (\$from\_user="", \$to\_user, \$title, \$message, \$object\_id="", \$site="")

## Input parameters

- \$from\_user ... from user name [string] (optional)
- \$to\_user ... to user name [string]
- \$title ... title [string]
- \$message ... message [string]
- \$object\_id ... object ID or object path [string] (optional)
- \$site ... publication name [string] (optional)

### global input parameters

- \$mgmt\_config
- \$hcms\_lang\_codepage
- \$hcms\_lang
- \$lang

#### Output

true/false

## 7.1.174 linking\_valid

## Description

Verifies if a valid access linking exists.

### **Syntax**

linking\_valid ()

#### Input parameters

## global input parameters

- \$mgmt\_config
- \$hcms\_linking

## Output

true/false

# 7.1.175 linking\_objects

### Description

Returns the all objects of an access link as an array.

#### **Syntax**

linking\_objects (\$return\_text\_id=array())

## Input parameters

• \$return\_text\_id ... %

#### global input parameters

- \$mgmt\_config
- \$hcms\_linking

### **Output**

• result array / false on error

## 7.1.176 linking\_inscope

#### **Description**

Verifies if the provided location path is in the scope of the access linking.

#### **Syntax**

linking\_inscope (\$site="", \$location, \$page="", \$cat="")

#### Input parameters

- \$site ... publication name [string] (optional)
- \$location ... location [string]
- \$page ... object name [string] (optional)
- \$cat ... object category [comp, page] (optional)

## global input parameters

- \$mgmt\_config
- \$hcms\_linking

#### Output

true/false

## 7.1.177 savecontent

### **Description**

Saves the provided content for a specific object. Only the provided content based on its ID will be saved. Existing content with a different ID will not be deleted.

```
Example of a page content array as content input:
array(
array(
"pagetitle" = > "My Page",
"pageauthor" = > "Thomas Tester",
"pagedescription"=>"Just an example"
),
array(
"id"=>"MyText",
"textu"=>"I am a text"
array(
"id"=>"MyImage",
"mediaobject" = > "%comp%/myPublication/image.jpg",
"mediaalttext"=>"This is my image",
"mediaalign"=>"top",
"mediawidth"=>"260"
"mediaheight"=>"140",
),
array(
"id"=>"MyLink",
"linkhref" = > "http://www.hypercms.com",
"linktarget"=>"_SELF",
"linktext"=>"hyperCMS Home"
),
array(
"id" = > "MyComponent",
"componentm"=>"%comp%/comp_1.html|%comp%/comp_2.html",
"condition"=>""
)
)
```

#### **Syntax**

savecontent (\$site, \$location, \$page, \$content, \$charset="UTF-8", \$user, \$db\_connect="")

## Input parameters

- \$site ... publication name [string]
- \$location ... location [string]
- \$page ... object name [string]
- \$content ... content array with 1st key as index number and 2nd keys as id or name according to the template tags [array]
- \$charset ... character set [string] (optional)
- \$user ... user name [string]
- \$db\_connect ... DB connectivity file name [string] (optional)

## global input parameters

- \$mgmt\_config
- \$lang
- \$hcms\_lang
- \$eventsystem

- \$rootpermission
- \$globalpermission
- \$localpermission
- \$siteaccess
- \$pageaccess
- \$compaccess
- \$adminpermission
- \$hiddenfolder

true/false

# 7.2 Get API Functions

## 7.2.1 getserverload

### **Syntax**

getserverload (\$interval=0)

## Input parameters

• \$interval ... intervall for average system load can be 1

## Output

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

## 7.2.2 getconfigvalue

#### Description

Help function for createinstance

#### **Syntax**

getconfigvalue (\$config, \$in\_key="")

## Input parameters

- \$config ... settings [array]
- \$in\_key ... value/substring in array key [string] (optional)

## Output

value of setting

## 7.2.3 getsession

### **Syntax**

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

## Input parameters

- \$variable ... session variable name [string]
- \$default ... default session value [string] (optional)

## Output

session value

## 7.2.4 getrequest

#### **Description**

Returns the value of a POST, GET or COOKIE request, or a default value if not valid.

#### Syntax

getrequest (\$variable, \$force\_type=false, \$default="")

## Input parameters

- \$variable ... request variable name [string]
- \$force\_type ... must be of certain type [numeric, array, publicationname, locationname, objectname, url, bool] (optional)
- \$default ... default value [string] (optional)

#### Output

request value

## 7.2.5 getrequest\_esc

#### **Description**

Returns the escaped value in order to prevent XSS from POST, GET or COOKIE variables. Returns a default value if not valid.

#### Syntax

getrequest\_esc (\$variable, \$force\_type=false, \$default="", \$js\_protection=false)

#### Input parameters

- \$variable ... request variable name [string]
- \$force\_type ... must be of certain type [numeric, array, publicationname, locationname, objectname] (optional)
- \$default ... default value [string] (optional)
- \$js\_protection ... remove characters to avoid JS injection [boolean] (optional)

request value

## 7.2.6 getuserip

## Description

Retrieves the IP address of the client/user.

# **Syntax**

getuserip ()

## Input parameters

#### Output

• IP address of client / false on error

## 7.2.7 getobjectlistcells

## **Syntax**

getobjectlistcells (\$viewportwidth, \$is\_mobile=0)

### Input parameters

- \$viewportwidth ... width of viewport or window in pixels [integer]
- \$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 [string] (optional)

## global input parameters

• \$mgmt\_config

## Output

• language file name

# 7.2.10 getcodepage

## **Syntax**

getcodepage (\$lang="en")

## Input parameters

• \$lang ... language code [string] (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 [string] (optional)

## global input parameters

• \$mgmt\_config

## Output

• supported language code for calendar

# 7.2.12 getescapedtext

## Description

Returns the synonyms of a word.

#### **Syntax**

getescapedtext (\$text, \$charset="", \$lang="")

## Input parameters

- \$text ... word [string]
- \$charset ... 2-digit language code [string] (optional)
- \$lang

#### global input parameters

• \$mgmt\_config

## Output

• array holding all synonyms including the provided word / false on error

## 7.2.13 getsearchhistory

#### **Syntax**

getsearchhistory (\$user="")

## Input parameters

• \$user ... user name [string] (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

### 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.

#### **Syntax**

gettaxonomy\_sublevel (\$site, \$lang="en", \$tax\_id="0")

## Input parameters

- \$site ... publication name [string]
- \$lang ... language code [string] (optional)
- \$tax\_id ... taxonomy parent ID [string] (optional)

### global input parameters

- \$mgmt\_config
- \$taxonomy

### Output

array holding all keywords of the next taxonomy level / false on error

## 7.2.15 gettaxonomy\_childs

### 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.

## **Syntax**

gettaxonomy\_childs (\$site="", \$lang="", \$expression, \$childlevels=1, \$id\_only=true)

## Input parameters

- \$site ... publication name [string] (optional)
- \$lang ... language code [string] (optional)
- \$expression ... taxonomy ID or expression or taxonomy path in the form %taxonomy%/publication-name or 'default'/language-code/taxonomy-ID/taxonomy-child-levels [string]
- \$childlevels ... taxonomy child levels [integer] (optional)
- \$id\_only ... only return taxonomy IDs without language and keyword information [boolean] (optional)

# global input parameters

- \$mgmt\_config
- \$taxonomy

## Output

array holding all taxonomy IDs / false on error

# 7.2.16 gethierarchy\_defintion

#### **Description**

Reads the metadata/content hierarchy defintion and returns a multidimensinal array.

#### **Syntax**

gethierarchy\_defintion (\$site, \$selectname="")

### Input parameters

- \$site ... publication name [string]
- \$selectname ... hierarchy name [string] (optional)

### global input parameters

• \$mgmt\_config

## Output

• hierarchy array in form of array[name][level][text-id][language] = label / false on error

# 7.2.17 gethierarchy\_sublevel

### Description

Returns sorted values of a metadata/content hierarchy level.

#### **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 [string]

## global input parameters

- \$mgmt\_config
- \$lang

 array holding all hierarchy URLs as key and text content or label as value / false on error

## 7.2.18 getkeywords

#### **Description**

Generates an array holding all keywords and the number as value and keyword ID as key.

## **Syntax**

getkeywords (\$site="")

## Input parameters

• \$site ... publication name [string] (optional)

## global input parameters

• \$mgmt\_config

#### Output

keywords as array / false on error

## 7.2.19 getmetakeywords

# Description

Generates a keyword list from a plain text. Stop word lists are defined in data/include/stopwords.inc.php

#### **Syntax**

getmetakeywords (\$text, \$language="en", \$charset="UTF-8")

# Input parameters

- \$text ... text [string]
- \$language ... language to be used for stop word list [de, en, ...] (optional)
- \$charset ... character set [string] (optional)

# global input parameters

• \$mgmt\_config

keywords as array /false on error

## 7.2.20 getmetadescription

#### **Description**

Generates a description from a text, to be used as meta information.

### **Syntax**

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

### Input parameters

- \$text ... text [string]
- \$charset

## Output

cleanded description of provided text /false on error

## 7.2.21 getgooglesitemap

#### **Description**

Generates a google sitemap xml-output

#### **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 name [string]
- \$dir ... directory path [string]
- \$url ... URL to directory [string]
- \$getpara ... GET parameters to use for new versions of the URL as array (optional)
- \$permalink ... permanent links text-ID to use for location [array] (optional)
- \$chfreq ... frequency of google scrawler [never, weekly, daily] (optional)
- \$prio ... priority [1 or less] (optional)
- \$ignore ... ignore file names [array] (optional)
- \$filetypes ... allowed file types [array] (optional)
- \$show\_freq ... include frequenzy tag [boolean] (optional)
- \$show\_prio ... include priority tag [boolean] (optional)

#### global input parameters

- \$mgmt\_config
- \$publ\_config

xml sitemap / false on error

## 7.2.22 getlistelements

## **Syntax**

getlistelements (\$list\_sourcefile)

## Input parameters

• \$list\_sourcefile ... content attribute value of list or 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="\r\n", \$template="")

## Input parameters

- \$location ... location [string]
- \$object ... object name (both optional if container is given) [string]
- \$container ... container name/ID or container content [string] (optional)
- \$seperator ... seperator of meta data fields [any string, array] (optional)
- \$template ... publication name/template name to extract label names [string] (optional)

#### global input parameters

\$mgmt\_config

#### Output

• string with all metadata from given object based on container / false

## 7.2.24 getmetadata\_multiobjects

### 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.

## **Syntax**

getmetadata\_multiobjects (\$multiobject\_array, \$user)

## Input parameters

- \$multiobject\_array ... converted path of multiple objects [array]
- \$user ... user name [string]

## global input parameters

- \$mgmt\_config
- \$siteaccess
- \$pageaccess
- \$compaccess
- \$hiddenfolder
- \$adminpermission
- \$localpermission

## Output

assoziatve array with all text content and meta data / false

# 7.2.25 getmetadata\_container

### Description

Extracts container, media, and metadata information of a container. This function is used for the presentation of metadata for objectlist views.

#### **Syntax**

getmetadata\_container (\$container\_id, \$text\_id\_array)

### Input parameters

- \$container\_id ... container ID [string]
- \$text\_id\_array ... array of text IDs [array]

## global input parameters

- \$mgmt\_config
- \$labels

#### Output

• assoziatve array with all text content and meta data / false

# 7.2.26 getobjectlist

### Description

Get all objects of a location. This is a simplified wrapper for function rdbms\_searchcontent.

#### **Syntax**

getobjectlist (\$site="", \$location="", \$folderhash="", \$objectlistcols=array())

## Input parameters

- \$site ... publication name [string] (optional)
- \$location ... location [string] (optional)
- \$folderhash ... folder hash code [string] (optional)
- \$objectlistcols ... text IDs to be returned e.g. text: Title [array] (optional)

### global input parameters

\$mgmt\_config

## Output

result array / false on error

## 7.2.27 getobjectcontainer

## Description

Loads the content container of a given object (page, component, folder).

#### **Syntax**

getobjectcontainer (\$site, \$location, \$object, \$user, \$type="work")

## Input parameters

- \$site ... publication name [string]
- \$location ... location [string]
- \$object ... object name [string]
- \$user ... user name [string]
- \$type ... container type [work, published] (optional)

#### global input parameters

\$mgmt\_config

### Output

• Content Container [XML]/false

# 7.2.28 getcontainer

## Description

Obsolete function used as an alias for the loadcontainer function without the possibility to load locked containers

## **Syntax**

getcontainer (\$containerid, \$type)

#### Input parameters

- \$containerid ... container name or container ID [string]
- \$type ... container type [published, work]

### global input parameters

• \$mgmt\_config

### Output

• Contant Container [XML]/false

# 7.2.29 getwallpaper

### Description

Provides a wallpaper image or video used for the logon and home screen.

#### Syntax

getwallpaper (\$theme="", \$version="")

## Input parameters

- \$theme ... design theme name [string] (optional)
- \$version ... version number for the wallpaper service [string] (optional)

## global input parameters

\$mgmt\_config

# Output

• URL of wallpaper image / false

## 7.2.30 getcontainername

## **Syntax**

getcontainername (\$container)

#### Input parameters

• \$container ... container name (e.g. 0000112.xml.wrk) or container ID [string]

### global input parameters

• \$mgmt\_config

## Output

 Array with file name of the working content container (locked or unlocked!) and username if locked

## 7.2.31 getlocationname

#### **Syntax**

getlocationname (\$site, \$location, \$cat, \$source="path")

#### Input parameters

- \$site ... publication name [string]
- \$location ... location path (as absolute path or converted path) [string]
- \$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 directory and file names / false on error

# 7.2.32 getthemes

### **Description**

Returns all design theme names as values and the techical names (path) as key of the result array.

#### **Syntax**

getthemes (\$site\_array=array())

## Input parameters

• \$site\_array ... publication name as string or array [string, array] (optional)

### global input parameters

\$mgmt\_config

## Output

• all design theme names as array / false

# 7.2.33 getthemelocation

## Description

Returns the absolute path (URL) of the theme (css and images).

If the main configuration setting \$mgmt\_config['theme'] defines a theme, this theme will be mandatory in case it exists.

## **Syntax**

getthemelocation (\$theme="")

## Input parameters

• \$theme ... theme name [string] (optional)

## global input parameters

• \$mgmt\_config

### **Output**

• path to theme / false

# 7.2.34 getcategory

#### **Description**

Evaluates the category ['page, comp'] of a location

#### Syntax

getcategory (\$site="", \$location)

#### Input parameters

- \$site ... publication name [string] (optional)
- \$location ... location path [string]

## global input parameters

- \$mgmt\_config
- \$publ\_config

- category ['page
- comp'] / false on error

# 7.2.35 getpublication

#### **Description**

Extract the publication name of a location path

## **Syntax**

getpublication (\$path)

## Input parameters

• \$path ... converted location path [string]

## Output

• publication name

# 7.2.36 getlocation

## Description

Extract the location excluding object or folder of a location path

### **Syntax**

getlocation (\$path)

## Input parameters

• \$path ... location path [string]

## Output

• location (without object or folder)

# 7.2.37 getobject

### **Description**

Extract the object or folder of a location path

#### **Syntax**

getobject (\$path)

## Input parameters

• \$path ... location path [string]

## Output

• object or folder name

# 7.2.38 getmediacontainername

## Description

Extract the container name from a multimedia file name by using the hcm-ID

#### **Syntax**

getmediacontainername (\$file)

## Input parameters

• \$file ... file name [string]

## Output

• container name / false on error

# 7.2.39 getmediafileversion

#### **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.

### **Syntax**

getmediafileversion (\$container)

#### Input parameters

• \$container ... container name or container ID [string]

## global input parameters

- \$mgmt\_config
- \$user

## Output

media file name / false on error

# 7.2.40 getobjectid

## Description

Converts the object path to the object ID of any object

#### **Syntax**

getobjectid (\$objectlink)

## Input parameters

• \$objectlink ... converted object path or pathes separated by | [string]

## Output

object ID

## 7.2.41 getobjectlink

## Description

Converts the object ID to the object path of any object

#### **Syntax**

getobjectlink (\$objectid)

#### Input parameters

• \$objectid ... converted object ID or IDs separated by | [string]

## Output

• converted object link

# 7.2.42 getcontainerversions

#### **Syntax**

getcontainerversions (\$container)

## Input parameters

• \$container ... container ID or container name [string]

## global input parameters

• \$mgmt\_config

## Output

• array of all versions (array[version-extension] = file-name) / false

# 7.2.43 getlocaltemplates

### **Description**

This function returns a list of all templates of a publication without inherited templates from other publications.

#### **Syntax**

getlocaltemplates (\$site, \$cat="all")

#### Input parameters

- \$site ... publication name [string]
- \$cat ... template category [all, page, comp, meta, inc, portal] (optional)

## global input parameters

• \$mgmt\_config

## Output

• array with all template names / false

## 7.2.44 gettemplates

#### 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.

Portal templates are not supoported by the template inheritance due to the fact that the portal access link permission is connected to a specific publication.

## **Syntax**

gettemplates (\$site, \$cat="all")

### Input parameters

- \$site ... publication name [string]
- \$cat ... object category [all, page, comp, meta] (optional)

#### global input parameters

- \$user
- \$mgmt\_config
- \$hcms\_lang
- \$lang

• template file name list as array / false on error

## 7.2.45 gettemplateversions

### **Syntax**

gettemplateversions (\$site, \$template)

## Input parameters

- \$site ... publication name [string]
- \$template ... template name [string]

### global input parameters

• \$mgmt\_config

## Output

array of all versions (array['YYYY-MM-DD HH: MM: SS'] = file-name) / false

## 7.2.46 getfileinfo

### 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 page or component is published (true) or not (false), true in all other

cases

\$result['deleted']: if file is deleted = true else = false

#### **Syntax**

getfileinfo (\$site, \$file, \$cat="comp")

#### Input parameters

- \$site ... publication name [string] (optional)
- \$file ... file name incl. extension [string]
- \$cat ... category [page, comp] (optional)

#### global input parameters

• \$mgmt\_config

• array/false

# 7.2.47 getobjectinfo

## Description

Get all file pointers (container, media, template) and object name from object file and collect info from container version, if provided

### **Syntax**

getobjectinfo (\$site, \$location, \$object, \$user="sys", \$container\_version="")

### Input parameters

- \$site ... publication name [string]
- \$location ... location [string]
- \$object ... object name [string]
- \$user ... user name [string] (optional)
- \$container\_version ... container version [string] (optional)

## global input parameters

• \$mgmt\_config

### Output

• result array / false on error

## 7.2.48 getfilesize

### Description

This function won't give you a proper result of the file size of multimedia components, if there is no Database installed.

## **Syntax**

getfilesize (\$file)

### Input parameters

• \$file ... converted path to file or directory [string]

### global input parameters

\$mgmt\_config

## Output

• result array with file size in kB and file count / false on error

# 7.2.49 getmimetype

### **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.

#### Syntax

getmimetype (\$file)

### Input parameters

• \$file ... file name incl. extension [string]

## global input parameters

• \$mgmt\_config

### **Output**

• mime\_type

## 7.2.50 getfiletype

#### **Syntax**

getfiletype (\$file\_ext)

#### Input parameters

• \$file\_ext ... file extension or file name [string]

### global input parameters

- \$mgmt\_config
- \$hcms\_ext

#### Output

• file type to be saved in database based on file extension

## 7.2.51 getpreviewwidth

#### **Description**

Returns the default preview/annotation width in pixel of a document, image, or video

## **Syntax**

getpreviewwidth (\$site, \$filepath, \$width\_orig="")

## Input parameters

- \$site ... publication name [string] (optional)
- \$filepath ... path to file or file name [string]
- \$width\_orig ... original width [string] (optional)

## global input parameters

- \$mgmt\_config
- \$hcms\_ext

## Output

• result array with width and height / false on error

# 7.2.52 getimagecolorkey

#### **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

### **Syntax**

getimagecolorkey (\$image)

## Input parameters

• \$image ... image resource [resource]

# global input parameters

• \$mgmt\_config

### Output

• color key of image / false on error

## 7.2.53 getimagecolors

## 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)

## **Syntax**

getimagecolors (\$site, \$file)

### Input parameters

- \$site ... publication name [string]
- \$file ... media file name [string]

## global input parameters

- \$mgmt\_config
- \$user

## Output

result array / false on error

# 7.2.54 getmediasize

#### **Syntax**

getmediasize (\$filepath)

## Input parameters

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

## global input parameters

- \$mgmt\_config
- \$mgmt\_imagepreview
- \$user

## Output

• Array with media width and height / false on error

# 7.2.55 getimageinfo

## **Syntax**

getimageinfo (\$filepath)

#### Input parameters

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

## global input parameters

- \$mgmt\_config
- \$mgmt\_imagepreview
- \$user

## Output

- Array with image information like md5 hash
- width
- height
- colors / false on error

# 7.2.56 getpdfinfo

## Description

Extracts width and height in pixel of a PDF file based on the MediaBox in the files content or ImageMagick as fallback

### **Syntax**

getpdfinfo (\$filepath, \$box="MediaBox")

## Input parameters

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

# 7.2.57 getvideoinfo

## Description

Extract video metadata from video file.

#### **Syntax**

getvideoinfo (\$filepath)

## Input parameters

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

## global input parameters

- \$mgmt\_config
- \$mgmt\_mediapreview
- \$user

## Output

• video file information as result array / false on error

## 7.2.58 getbrowserinfo

## **Syntax**

getbrowserinfo ()

## Input parameters

### Output

• client browser and version as array / false on error

## 7.2.59 getcontentlocation

#### **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.

#### **Syntax**

getcontentlocation (\$container\_id, \$type="abs\_path\_content")

## Input parameters

- \$container\_id ... container id [string]
- \$type ... type [url\_path\_content, abs\_path\_content]

# global input parameters

\$mgmt\_config

## Output

• location of the container file / false on error

# 7.2.60 getmedialocation

### 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.

If the file resides outside the repository (symbolic link is used in the repository), the full path including the file name can be returned.

## **Syntax**

getmedialocation (\$site, \$file, \$type, \$resolve\_symlink=false)

### Input parameters

- \$site ... publication name [string]
- \$file ... multimedia file name (including hcm-ID) [string]
- \$type ... type [url\_path\_media, abs\_path\_media, url\_publ\_media, abs\_publ\_media]
- \$resolve\_symlink ... resolve symbolik links [boolean] (optional)

## global input parameters

- \$mgmt\_config
- \$publ\_config

#### **Output**

location of the multimedia file / false on error

## 7.2.61 getlockedfileinfo

#### **Description**

Finds the locked file and returns the name and user as array

# **Syntax**

getlockedfileinfo (\$location, \$file)

## Input parameters

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

#### global input parameters

\$mgmt\_config

### Output

• Array holding file name incl. lock extension and user name / false on error

# 7.2.62 getfavorites

## **Syntax**

getfavorites (\$user, \$output="path", \$return\_text\_id=array())

## Input parameters

- \$user ... user name [string]
- \$output ... output format [path, id] (optional)
- \$return\_text\_id ... text IDs to be returned if output=path [array] (optional)

### global input parameters

\$mgmt\_config

## Output

• object info or object id array of users favorites / false

## 7.2.63 gethomeboxes

### **Syntax**

gethomeboxes (\$site\_array=array())

# Input parameters

• \$site\_array ... pupblication name [array] (optional)

## global input parameters

• \$mgmt\_config

## Output

 All home boxes as array with technical name as key and readable name as value / false

## 7.2.64 getuserboxes

#### **Syntax**

getuserboxes (\$user)

#### Input parameters

• \$user ... user name [string]

## global input parameters

• \$mgmt\_config

## Output

• selected home box of a user as array with technical name as key and readable name as value / false

## 7.2.65 getusersonline

#### **Syntax**

getusersonline (\$sites=array())

## Input parameters

• \$sites ... publication names [array] (optional)

## global input parameters

- \$mgmt\_config
- \$siteaccess

## Output

• Array of online user names / false

# 7.2.66 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

# 7.2.67 getimagelib

## **Syntax**

getimagelib ()

#### Input parameters

## global input parameters

• \$mgmt\_imagepreview

## Output

- name of image library used [GD]
- ImageMagick] / false on error

## 7.2.68 getfilename

### Description

Extracts the file name of the content and template pointer tags of an object file

### Syntax

getfilename (\$filedata, \$tagname)

## Input parameters

- \$filedata ... file content [string]
- \$tagname ... hyperCMS tag name in page or component [string]

## Output

• file name

## 7.2.69 gethypertag

#### **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.

#### 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]

# Output

• full hyperCMS tag array [array]/false on error

# 7.2.70 gethypertagname

#### **Description**

Reads the name of the hyperCMS tag

#### **Syntax**

gethypertagname (\$tagdata)

### Input parameters

\$tagdata ... full hyperCMS tag [string]

#### **Output**

• full hyperCMS tag name/false on error

# 7.2.71 gethtmltag

### 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!

### **Syntax**

gethtmltag (\$filedata, \$tag)

### Input parameters

- \$filedata ... file content [string]
- \$tag ... full hyperCMS tag (or other identifier) [string]

### Output

• full html tag/false on error

# 7.2.72 gethtmltags

#### **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></a>, <div></div> and so on.

### **Syntax**

gethtmltags (\$filedata, \$tag)

### Input parameters

- \$filedata ... file content [string]
- \$tag ... full hyperCMS tag or other identifier in html tag [string]

### Output

string from html tag start to end tag/false on error

## 7.2.73 getattribute

### Description

Get the value of a certain attribute out of a string (...attributname=value....)

#### **Syntax**

getattribute (\$string, \$attribute, \$secure=true)

### Input parameters

- \$string ... string including attributes [string]
- \$attribute ... attribute name [string]
- \$secure ... secure attribute value reg. XSS [boolean] (optional)

### Output

• attribute value/false on error

## 7.2.74 getoption

### Description

Get the value of a certain option out of a string (-c:v value -ar 44100)

### **Syntax**

getoption (\$string, \$option)

### Input parameters

- \$string ... string including options [string]
- \$option ... option name [string]

### Output

• option value / false on error

## 7.2.75 getcharset

### Description

Extract the content-type definition and the character set from the template (1st priority), content container (2nd priority) or publication settings (3rd priority)

### **Syntax**

getcharset (\$site, \$data)

### Input parameters

- \$site ... publication name [string]
- \$data ... data from template or content container [string]

# global input parameters

\$mgmt\_config

### Output

array with content-type and charset / false on error

# 7.2.76 getartid

## Description

Extract the article ID from the tag ID

### **Syntax**

getartid (\$id)

### Input parameters

• \$id ... string including id [string]

### Output

article id/false on error

## 7.2.77 getelementid

### Description

Extract the element ID from the tag ID

#### Syntax

getelementid (\$id)

### Input parameters

• \$id ... string including id [string]

### Output

· element id/false on error

# 7.2.78 getfirstkey

### **Syntax**

getfirstkey (\$array)

### Input parameters

• \$array ... array [array]

### Output

• array key of first element in array if \$value is not empty / false on error

# 7.2.79 getdirectoryfiles

## **Syntax**

getdirectoryfiles (\$dir, \$pattern="")

### Input parameters

- \$dir ... path to directory [string]
- \$pattern ... pattern as string [string] (optional)

## Output

• sorted array of all files matching the pattern / false on error

# 7.2.80 getuserinformation

#### Description

This function creates an assoziative array with user information, e.g. for a user select box.

#### **Syntax**

getuserinformation (\$login="")

### Input parameters

• \$login ... user name [string] (optional)

### global input parameters

\$mgmt\_config

• \$user

### Output

 assoziative array with basic user information [publication->username->attribute] / false

# 7.2.81 getworkflowitem

### **Syntax**

getworkflowitem (\$site, \$workflow\_file, \$workflow, \$user)

#### Input parameters

- \$site ... publication name [string]
- \$workflow\_file ... location [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 [string or array]
- \$content ... value [string or array] (optional)
- \$write ... write session data for load balancer [boolean] (optional)

### Output

• true / false on error

### 7.3.2 settemplate

### Description

This function sets the template for a single folder/object or all objects in a folder.

#### **Syntax**

settemplate (\$site, \$location, \$object, \$template, \$recursive=false)

### Input parameters

- \$site ... publication name [string]
- \$location ... location [string]
- \$object ... object [string]
- \$template ... template name [string]
- \$recursive ... recursive [boolean] (optional)

### global input parameters

\$mgmt\_config

### Output

true/false

# 7.3.3 settaxonomy

#### **Description**

Analyzes the content regarding all taxonomy keywords, saves results in database and returns an array (multilingual support based on taxonomies).

### **Syntax**

settaxonomy (\$site, \$container\_id, \$langcode="", \$taxonomy="")

### Input parameters

- \$site ... publication name [string]
- \$container\_id ... container ID [string]
- \$langcode ... 2-digit language code [string] (optional)
- \$taxonomy ... taxonomy definition [array] (optional)

## global input parameters

\$mgmt\_config

### Output

result array / false on error

### 7.3.4 setarticle

### **Description**

Set article content in container. The content container will be returned and not saved.

#### **Syntax**

setarticle (\$site, \$contentdata, \$contentfile, \$arttitle=array(), \$artstatus=array(), \$artdatefrom=array(), \$artdateto=array(), \$artuser=array(), \$user="sys")

#### Input parameters

- \$site ... publication name [string]
- \$contentdata ... container (XML) [string]
- \$contentfile ... container name [string]
- \$arttitle ... article title [array]
- \$artstatus ... article status [array]
- \$artdatefrom ... article beginn date [array] (optional)
- \$artdateto ... article end date [array] (optional)
- \$artuser ... article user name [array or string] (optional)
- \$user ... user name [string] (optional)

### global input parameters

\$mgmt\_config

#### **Output**

- updated content container (XML)
- false on error

#### 7.3.5 settext

#### **Description**

Set text content in container and database. The content container will be returned and not saved.

#### **Syntax**

settext (\$site, \$contentdata, \$contentfile, \$text=array(), \$type=array(), \$art="no", \$textuser=array(), \$user="sys", \$charset="", \$addmicrotime=false)

### Input parameters

- \$site ... publication name [string]
- \$contentdata ... container (XML) [string]
- \$contentfile ... container name [string]
- \$text ... text with tag Id as key and text as value [array]
- \$type ... text type [array or string] [u, f, l, c, d, k]
- \$art ... article [array or string] [yes, no] (optional)
- \$textuser ... text user [array or string] (optional)
- \$user ... user name [string] (optional)
- \$charset ... character set of text content [string] (optional)
- \$addmicrotime ... add microtime to ID used for comments [boolean] (optional)

- \$mgmt\_config
- \$publ\_config

### **Output**

- updated content container (XML)
- false on error

## 7.3.6 setmedia

### Description

Set media content in container and database. The content container will be returned and not saved.

#### **Syntax**

setmedia (\$site, \$contentdata, \$contentfile, \$mediafile=array(), \$mediaobject=array(), \$mediaalttext=array(), \$mediaalign=array(), \$mediawidth=array(), \$mediaheight=array(), \$art="no", \$mediauser="", \$user="sys", \$charset="")

#### Input parameters

- \$site ... publication name [string]
- \$contentdata ... container (XML) [string]
- \$contentfile ... container name [string]
- \$mediafile ... media files with tag ID as key and reference as value [array] (optional)
- \$mediaobject ... new media object references with tag ID as key and reference as value [array]
- \$mediaalttext ... media alternative text [array] (optional)
- \$mediaalign ... media alignment [array] (optional)
- \$mediawidth ... media width [array] (optional)
- \$mediaheight ... media height [array] (optional) \$art ... article [array or string] [yes, no] (optional)
- \$mediauser ... content user [array or string] (optional)
- \$user ... user name [string] (optional)
- \$charset ... character set of text content [string] (optional)

# global input parameters

• \$mgmt\_config

### **Output**

- updated content container (XML)
- false on error

### 7.3.7 setpagelink

### **Description**

Set link content in container and database. The content container will be returned and not saved.

#### **Syntax**

setpagelink (\$site, \$contentdata, \$contentfile, \$linkhref=array(), \$linktarget=array(),
\$linktext=array(), \$art="no", \$linkuser=array(), \$user="sys", \$charset="")

#### Input parameters

- \$site ... publication name [string]
- \$contentdata ... container (XML) [string]
- \$contentfile ... container name [string]
- \$linkhref ... new link with tag ID as key and link reference as value [array]
- \$linktarget ... link target [array] (optional)
- \$linktext ... link text [array] (optional)
- \$art ... article [array or string] [yes, no] (optional)
- \$linkuser ... content user [array or string] (optional)
- \$user ... user name [string] (optional)
- \$charset ... character set of text content [string] (optional)

### global input parameters

• \$mgmt\_config

# Output

- updated content container (XML)
- false on error

#### 7.3.8 setcomplink

#### Description

Set component link content in container and database. The content container will be returned and not saved.

#### **Syntax**

setcomplink (\$site, \$contentdata, \$contentfile, \$component=array(), \$condition=array(),
\$art="no", \$compuser=array(), \$user="sys")

#### Input parameters

- \$site ... publication name [string]
- \$contentdata ... container (XML) [string]
- \$contentfile ... container name [string]
- \$component ... new components with tag ID as key and component reference as value [array]
- \$condition ... conditions [array] (optional)
- \$art ... article [array or string] [yes, no] (optional)
- \$compuser ... content user [array or string] (optional)
- \$user ... user name [string] (optional)

\$mgmt\_config

### Output

- updated content container (XML)
- false on error

### 7.3.9 sethead

#### **Description**

Only used for content in general head information of container.

### **Syntax**

sethead (\$site, \$contentdata, \$contentfile, \$headcontent=array(), \$user="sys", \$charset="")

#### Input parameters

- \$site ... publication name [string]
- \$contentdata ... container (XML) [string]
- \$contentfile ... container name [string]
- \$headcontent ... head content with tagname as ID and text as value [array]
- \$user ... user name [string] (optional)
- \$charset ... character set of text content [string] (optional)

### global input parameters

• \$mgmt\_config

### Output

- updated content container (XML)
- false on error

### 7.3.10 setrelation

### Description

This function sets a relationship between two objects by adding the reference as a multi component link to the specified tag ID of both objects.

#### **Syntax**

setrelation (\$site, \$location\_1, \$object\_1, \$id\_1="Related", \$location\_2, \$object\_2, \$id\_2="Related", \$user)

#### Input parameters

• \$site ... publication name [string]

- \$location\_1 ... location path 1 [string]
- \$object\_1 ... object name 1 for component link reference 2 [string]
- \$id\_1 ... tag/content ID 1 for component reference to object 2 [string] (optional)
- \$location\_2 ... location path 2 [string]
- \$object\_2 ... object name 2 for component link reference 2 [string]
- \$id\_2 ... tag/content ID 2 for component reference to object 1 [string] (optional)
- \$user ... user name [string]

\$mgmt\_config

## Output

true / false on error

#### 7.3.11 setfilename

#### Description

Sets or creates the file name of the hyperCMS content file, template file, media file or file name pointer

### Syntax

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

#### Input parameters

- \$filedata ... file content [string]
- \$tagname ... hyperCMS tag name in page or component [content, template, media, name]
- \$value ... new value [string]

### Output

• filedata/false on error

### 7.3.12 setuserboxes

#### Syntax

setuserboxes (\$name\_array, \$user)

#### Input parameters

- \$name\_array ... home box file names [array or string]
- \$user ... user name [string]

#### global input parameters

• \$mgmt\_config

## Output

true / false

# 7.3.13 setguiview

### **Syntax**

setguiview (\$objectview, \$explorerview, \$sidebar, \$user)

### Input parameters

- \$objectview ... object view name [formedit, cmsview, inlineview]
- \$explorerview ... explorer view name [detail, small, medium, large]
- \$sidebar ... show sidebar [true=1, false=0]
- \$user ... user name [string]

### global input parameters

• \$mgmt\_config

### Output

true / false

## 7.4 Connect API Functions

### 7.4.1 ftp\_userlogon

#### Description

This function connects and performs logon to an FTP server.

#### **Syntax**

ftp\_userlogon (\$server, \$user, \$passwd, \$ssl=false)

#### Input parameters

- \$server ... FTP servername or IP [string]
- \$user ... user name [string]
- \$passwd ... password [string]
- \$ssl ... SSL [boolean] (optional)

### global input parameters

• \$mgmt\_config

### Output

• true / false on error

# 7.4.2 ftp\_userlogout

### Description

This function disconnects from an FTP server.

### **Syntax**

ftp\_userlogout (\$conn\_id)

### Input parameters

• \$conn\_id ... FTP connection [resource]

### global input parameters

• \$mgmt\_config

# Output

• true / false on error

# 7.4.3 ftp\_getfile

### Description

This function gets a file from the FTP server.

#### Syntax

ftp\_getfile (\$conn\_id, \$remote\_file, \$local\_file, \$passive=true)

### Input parameters

- \$conn\_id ... FTP connection [resource]
- \$remote\_file ... path to file on FTP server [string]
- \$local\_file ... passive mode [boolean] (optional)
- \$passive

### global input parameters

• \$mgmt\_config

### Output

true / false on error

## 7.4.4 ftp\_putfile

### Description

This function puts a file to the FTP server.

#### **Syntax**

ftp\_putfile (\$conn\_id, \$local\_file, \$remote\_file, \$passive=true)

### Input parameters

- \$conn\_id ... FTP connection [resource]
- \$local\_file ... path to local file [string]
- \$remote\_file ... path to file on FTP server [string]
- \$passive ... passive mode [boolean] (optional)

### global input parameters

• \$mgmt\_config

### Output

• true / false on error

## 7.4.5 ftp\_deletefile

### Description

This function deletes a file from the FTP server.

### **Syntax**

ftp\_deletefile (\$conn\_id, \$remote\_file, \$passive=true)

## Input parameters

- \$conn\_id ... FTP connection [resource]
- \$remote\_file ... path to file on FTP server [string]
- \$passive ... passive mode [boolean] (optional)

### global input parameters

• \$mgmt\_config

#### Output

• true / false on error

## 7.4.6 ftp\_filelist

### Description

This function gets a file/directory listing of the FTP server.

### **Syntax**

ftp\_filelist (\$conn\_id, \$path=".", \$passive=true)

### Input parameters

- \$conn\_id ... FTP connection [resource]
- \$path ... path to remote directory [string] (optional)
- \$passive ... passive mode [boolean] (optional)

# global input parameters

\$mgmt\_config

### Output

• result array / false on error

## 7.4.7 createsharelink\_facebook

### **Syntax**

createsharelink\_facebook (\$site, \$url)

### Input parameters

- \$site ... URL to share [string]
- \$url

### global input parameters

\$mgmt\_config

### Output

• Share URL / false on error

### 7.4.8 createsharelink\_twitter

### **Syntax**

createsharelink\_twitter (\$site, \$url, \$text)

### Input parameters

• \$site ... URL to share [string]

- \$url ... message to share [string]
- \$text

• \$mgmt\_config

### Output

• Share URL / false on error

## 7.4.9 createsharelink\_googleplus

## **Syntax**

createsharelink\_googleplus (\$site, \$url)

#### Input parameters

- \$site ... URL to share [string]
- \$url

### global input parameters

• \$mgmt\_config

### Output

• Share URL / false on error

## 7.4.10 createsharelink\_linkedin

### **Syntax**

createsharelink\_linkedin (\$site, \$url, \$title, \$summary, \$source)

### Input parameters

- \$site ... URL to share [string]
- \$url ... title [string]
- \$title ... summary [string] (optional)
- \$summary ... source [string] (optional)
- \$source

# global input parameters

• \$mgmt\_config

### Output

• Share URL / false on error

## 7.4.11 createsharelink\_pinterest

### **Syntax**

createsharelink\_pinterest (\$site, \$image\_url, \$title, \$description)

### Input parameters

- \$site ... image URL to share [string]
- \$image\_url ... title [string]
- \$title ... description [string] (optional)
- \$description

### global input parameters

• \$mgmt\_config

### Output

• Share URL / false on error

# 7.5 Security API Functions

### 7.5.1 rootpermission

#### Description

Deseralizes the permission string and and returns the root permission array

### **Syntax**

rootpermission (\$site\_name, \$site\_admin, \$permission\_str)

#### Input parameters

- \$site\_name ... publication name [string]
- \$site\_admin ... publication admin [boolean]
- \$permission\_str ... permission string from group [string]

### global input parameters

- \$rootpermission
- \$mgmt\_config

### Output

· global permission array/false

### 7.5.2 global permission

#### **Description**

Deseralizes the permission string and returns the global permission array

#### **Syntax**

globalpermission (\$site\_name, \$permission\_str)

### Input parameters

- \$site\_name ... publication name [string]
- \$permission\_str ... permission string from group [string]

## Output

· global permission array/false

### 7.5.3 localpermission

### **Description**

Deseralizes the permission string and returns the local permission array

#### Syntax

localpermission (\$site\_name, \$permission\_str)

### Input parameters

- \$site\_name ... publication name [string]
- \$permission\_str ... permission string from group [string]

#### Output

• local permission array/false

### 7.5.4 accessgeneral

#### **Description**

Checks general access to certain system folders, publications and returns true if access is granted

#### **Syntax**

accessgeneral (\$site, \$location, \$cat)

### Input parameters

• \$site ... publication name [string]

- \$location ... location (path to folder) [string]
- \$cat ... object category [page, comp]

- \$mgmt\_config
- \$hiddenfolder
- \$siteaccess

### Output

true/false

# 7.5.5 accesspermission

#### **Description**

Evaluates page and component access permissions and returns group(s). Since version 8.0.0 this function does not evaluate the access based on access links anymore since explorer\_objectlist verifies the access linking.

### **Syntax**

accesspermission (\$site, \$location, \$cat)

### Input parameters

- \$site ... location (path to folder) [string]
- \$location ... object category [page, comp]
- \$cat

### global input parameters

- \$user
- \$pageaccess
- \$compaccess
- \$hiddenfolder
- \$hcms\_linking
- \$mgmt\_config

### Output

• group with access permissions as array / false on error

## 7.5.6 setlocal permission

### **Description**

Sets local permissions of a user group for a specific publication

### **Syntax**

setlocalpermission (\$site, \$group\_array, \$cat)

#### Input parameters

- \$site ... publication name [string]
- \$group\_array ... group name [array]
- \$cat ... object category [page, comp]

### global input parameters

\$localpermission

# Output

local permission array / false on error

## 7.5.7 checkpublicationpermission

#### Description

Checks the access to a publication based on the site access and inheritance settings

# **Syntax**

checkpublicationpermission (\$site, \$strict=true)

### Input parameters

- \$site ... publication name [string]
- \$strict ... strictly limited to siteaccess only without inheritance [boolean] (optional)

### global input parameters

- \$mgmt\_config
- \$siteaccess

### Output

 "direct" for direct access via group permission / "inherited" for access through inheritance / false

### 7.5.8 checkadminpermission

#### Description

Checks the super admin permission

#### Syntax

checkadminpermission ()

### Input parameters

• \$adminpermission

### Output

true/false

# 7.5.9 checkrootpermission

### Description

Checks the root permissions

#### **Syntax**

checkrootpermission (\$name)

### Input parameters

• \$name ... permission name [string]

### global input parameters

• \$rootpermission

### Output

• true/false

# 7.5.10 checkglobalpermission

## Description

Checks global permission for a publication

#### Syntax

checkglobalpermission (\$site, \$name)

### Input parameters

- \$site ... publication name [string]
- \$name ... permission name [string]

### global input parameters

• \$globalpermission

### Output

true/false

# 7.5.11 checklocalpermission

#### Description

Checks local permissions of a user group for a specific publication

#### **Syntax**

checklocalpermission (\$site, \$group, \$name)

### Input parameters

- \$site ... publication name [string]
- \$group ... user group name [string]
- \$name ... permission name [string]

## global input parameters

• \$\$localpermission

### Output

true/false

## 7.5.12 checklanguage

## **Syntax**

checklanguage (\$language\_array, \$language\_value)

### Input parameters

- \$language\_array ... language array with all valid values [array]
- \$language\_value ... language value of attribute in template tag [string}

## Output

• true if language array holds the given language value / false if not found

# 7.5.13 checkgroupaccess

### Description

Verifies if a user has access to the tags content based on the group membership.

#### Syntax

checkgroupaccess (\$groupaccess, \$usergroup\_array)

### Input parameters

- \$groupaccess ... group access from template group-tag attribute [string]
- \$usergroup\_array ... user group membership names [array]

### Output

• true if the current user group has access / false if not

### 7.5.14 userlogin

#### **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.

#### **Syntax**

```
userlogin ($user="", $passwd="", $hash="", $objref="", $objcode="",
$ignore_password=false, $locking=true, $portal="")
```

### Input parameters

- \$user ... user name [string] (optional if hash code is used for logon)
- \$passwd ... password [string] (optional if hash code is used for logon)
- \$hash ... hash code of user [string] (optional)
- \$objref ... object reference for hcms linking (object ID) [string] (optional)
- \$objcode ... object code for hcms linking (crypted object ID) [string] (optional)
- \$ignore\_password ... ignore passwordcheck needed for WebDAV or access link [boolean] (optional)
- \$locking ... lock IP after 10 failed attempts to login [boolean] (optional)
- \$portal ... portal name in the form of publication.portal or publication/portal [string] (optional)

#### global input parameters

- \$mgmt\_config
- \$eventsystem
- \$hcms\_lang\_codepage
- \$hcms\_lang
- \$lang

### Output

result array

## 7.5.15 setportalpermissions

#### **Description**

Sets the permissions for a portal user by reducing the standard permissions.

#### **Syntax**

setportalpermissions (\$login\_result)

### Input parameters

\$login\_result ... result from function userlogin [array]

### global input parameters

• \$mgmt\_config

### Output

result array / false

### 7.5.16 registerinstance

### **Description**

Registers the instance in the users session.

#### Syntax

registerinstance (\$instance, \$load\_config=true)

### Input parameters

- \$instance ... instance name [string]
- \$load\_config ... load main config of instance [boolean] (optional)

### global input parameters

• \$mgmt\_config

### Output

true/false

## 7.5.17 registeruser

#### **Description**

Registers all user related paramaters in the session. Access links can be provided with the login result or alternatively as the seperate accesslink parameter.

### **Syntax**

registeruser (\$instance="", \$login\_result, \$accesslink=false, \$hcms\_objformats=false, \$is\_mobile=0, \$is\_iphone=0, \$html5support=1)

### Input parameters

- \$instance ... instance name [string] (optional)
- \$login\_result ... result array of function userlogin [array]

- \$accesslink ... access link [array] (optional)
- \$hcms\_objformats ... download formats of access link provided by function rdbms\_getaccessinfo [array] (optional)
- \$is\_mobile ... mobile browser result of client [0, 1] (optional)
- \$is\_iphone ... is iOS browser result of client [0, 1] (optional)
- \$html5support ... HTML5 file support result of client [0, 1] (optional)

- \$mgmt\_config
- \$hcms\_lang
- \$lang

### Output

• result array / false on error

# 7.5.18 registerassetbrowser

#### **Syntax**

registerassetbrowser (\$userhash, \$objecthash="")

### Input parameters

- \$userhash ... user hash [string]
- \$objecthash ... object hash [string] (optional)

#### global input parameters

• \$mgmt\_config

### Output

true/false

#### 7.5.19 createchecksum

#### Description

Creates the checksum of the user permissions

#### **Syntax**

createchecksum (\$permissions="")

# Input parameters

• \$permissions ... array or empty [array]

### Output

• MD5 checksum

### 7.5.20 writesession

### Description

Writes hyperCMS specific session data of a user

### **Syntax**

writesession (\$user, \$passwd, \$checksum, \$siteaccess=array())

### Input parameters

- \$user ... user name [string]
- \$passwd ... password [string]
- \$checksum ... checksum [string]
- \$siteaccess ... publicaion access [array]

## global input parameters

• \$mgmt\_config

## Output

• true / false on error

## 7.5.21 writesessiondata

#### **Description**

Serializes and writes all session data of a user

#### **Syntax**

writesessiondata ()

#### Input parameters

### global input parameters

• \$mgmt\_config

### Output

• true / false on error

### 7.5.22 createsession

### Description

Checks if session data of a user is available. This function does access session variables directly!

### **Syntax**

createsession ()

### Input parameters

### global input parameters

• \$mgmt\_config

## Output

true

#### 7.5.23 killsession

### Description

Destroys session data of a user

### **Syntax**

killsession (\$user="", \$destroy\_php=true, \$remove=false)

## Input parameters

- \$user ... user name for hyperCMS session [string] (optional)
- \$destroy\_php ... destroy php session [boolean] (optional)
- \$remove ... kill all sessions of the user [boolean] (optional)

### global input parameters

• \$mgmt\_config

### Output

true

# 7.5.24 checkdiskkey

### Description

Checks the disc key of the installation.

### **Syntax**

checkdiskkey (\$users="", \$site="")

### Input parameters

- \$users ... users XML [string] (optional)
- \$site ... publication names (use | as seperator) [string] (optional)

### global input parameters

• \$mgmt\_config

### Output

true / false

## 7.5.25 checkpassword

#### **Description**

This function checks the strength of a password and return the error messages or true

### **Syntax**

checkpassword (\$password, \$user="")

### Input parameters

- \$password ... password [string]
- \$user ... user name [string] (optional for password history)

### global input parameters

- \$mgmt\_config
- \$lang

#### **Output**

true if passed / error message as string

## 7.5.26 loguserip

#### **Syntax**

loguserip (\$client\_ip, \$user="sys")

### Input parameters

- \$client\_ip ... client IP address [string]
- \$user ... user logon name [string] (optional)

### global input parameters

• \$mgmt\_config

# Output

• true / false on error

# 7.5.27 checkuserip

# **Syntax**

checkuserip (\$client\_ip, \$user="", \$timeout=0)

### Input parameters

- \$client\_ip ... client IP address [string]
- \$user ... user logon name [string] (optional)
- \$timeout ... timeout in minutes [integer] (optional)

### global input parameters

• \$mgmt\_config

## Output

• true if IP is not locked / false if IP is locked or on error

# 7.5.28 checkuserrequests

#### Description

Provides security for Cross-Site Request Forgery

# **Syntax**

checkuserrequests (\$user="sys")

## Input parameters

• \$user ... user name [string] (optional)

#### global input parameters

• \$mgmt\_config

### Output

• true / false if a certain amount of reguests per minute is exceeded

### 7.5.29 checkusersession

### **Description**

Checks if the session data of a user is valid. This function does access session variables directly.

## **Syntax**

checkusersession (\$user="sys", \$CSRF\_detection=true)

#### Input parameters

- \$user ... user name [string] (optional)
- \$CSRF\_detection ... include CSRF detection [boolean]

# global input parameters

\$mgmt\_config

### Output

• true / html-output followed by termination requires config.inc.php

# 7.5.30 allowuserip

### Description

Verifies if the client IP is in the range of valid IPs and logs IP addresses with no access

### **Syntax**

allowuserip (\$site)

### Input parameters

• \$site ... publication name [string]

### global input parameters

• \$mgmt\_config

### Output

• true / false requires config.inc.php

# 7.5.31 valid\_objectname

#### **Description**

Checks if an object name includes forbidden characters in order to prevent directory browsing.

### **Syntax**

valid\_objectname (\$variable)

### Input parameters

• \$variable ... variable [string or array]

### Output

true / false

### 7.5.32 valid\_locationname

### Description

Checks if an location includes forbidden characters in order to prevent directory browsing

#### **Syntax**

valid\_locationname (\$variable)

### Input parameters

• \$variable ... variable [string or array]

### Output

true / false

## 7.5.33 valid\_publicationname

#### Description

Checks if a publication name includes forbidden characters in order to prevent directory browsing.

Optionally verifies if the publication name is included in the siteaccess variable.

## **Syntax**

valid\_publicationname (\$variable)

#### Input parameters

• \$variable ... variable [string or array]

### global input parameters

\$siteaccess

### Output

• true / false

## 7.5.34 html\_encode

### **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.

### **Syntax**

html\_encode (\$expression, \$encoding="", \$js\_protection=false)

### Input parameters

- \$expression ... variable [string or array]
- \$encoding ... conversion of all special characters based on given character set or to ASCII [string] (optional)
- \$js\_protection ... remove characters to avoid JS injection [boolean] (optional)

## Output

html encoded value as array or string / false on error

### 7.5.35 html\_decode

## Description

This function decodes all characters which have been converted by html\_encode

#### **Syntax**

html\_decode (\$expression, \$encoding="")

#### Input parameters

- \$expression ... epxression [string or array]
- \$encoding ... conversion of all special characters based on character set [string] (optional)

### **Output**

html decoded value as array or string / false on error

### 7.5.36 scriptcode\_encode

#### 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.

### **Syntax**

scriptcode\_encode (\$content)

### Input parameters

• \$content ... content [string]

### global input parameters

• \$mgmt\_config

### Output

escaped content as string / false on error

# 7.5.37 scriptcode\_extract

#### **Description**

This function extracts the script code of a given content

#### **Syntax**

scriptcode\_extract (\$content, \$identifier\_start="<?", \$identifier\_end="?>")

#### Input parameters

- \$content ... content [string]
- \$identifier\_start ... identifier of script begin [string]
- \$identifier\_end ... identifier of script end [string]

#### **Output**

• script code as array / false on error or if noting was found

### 7.5.38 scriptcode\_clean\_functions

#### **Description**

This function removes all dangerous PHP functions

#### Syntax

scriptcode\_clean\_functions (\$content, \$type=4, \$application="PHP")

## Input parameters

- \$content ... content [string]
- \$type ... cleaning level type: no cleaning = 0; basic set of disabled functions = 1; 1 + file access functions = 2; 2 + include functions = 3; 3 + hyperCMS API file functions = 4; No server side script allowed = 5 [0, 1, 2, 3, 4, 5] (optional)
- \$application ... application [PHP, ASP, JSP] (optional)

• \$mgmt\_config

### Output

• result array / false on error

# 7.5.39 sql\_clean\_functions

### Description

This function checks SQL statements for write operations

# **Syntax**

sql\_clean\_functions (\$content)

### Input parameters

• \$content ... SQL statement [string]

### global input parameters

• \$mgmt\_config

### Output

• result array / false on error

## 7.5.40 url\_encode

## Description

This function encodes all characters

### **Syntax**

url\_encode (\$variable)

### Input parameters

• \$variable ... variable [string or array]

### global input parameters

\$mgmt\_config

#### Output

• urlencoded value as array or string / false on error

# 7.5.41 url\_decode

### **Description**

This function decodes all characters which have been converted by url\_encode or urlencode (PHP)

#### **Syntax**

url\_decode (\$variable)

### Input parameters

• \$variable ... variable [string or array]

### global input parameters

• \$mgmt\_config

### Output

urldecoded value as array or string / false on error

### 7.5.42 shellcmd\_encode

#### **Description**

This function encodes/escapes characters to secure the shell comand.

#### **Syntax**

shellcmd\_encode (\$variable, \$type="")

#### Input parameters

- \$variable ... variable [string or array]
- \$type ... type [%, strict] (optional)

### Output

• encoded value as array or string / false on error

# 7.5.43 hcms\_crypt

### **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 for password hashing.

#### **Syntax**

hcms\_crypt (\$string, \$start=0, \$length=0)

#### Input parameters

- \$string ... string to encode [string]
- \$start ... start position [integer]
- \$length ... length for string extraction [integer]

# global input parameters

• \$mgmt\_config

### Output

• encoded string / false on error

## 7.5.44 hcms\_encrypt

#### **Description**

Encryption of a string. Only strong encryption is binary-safe!

#### **Syntax**

hcms\_encrypt (\$string, \$key="", \$crypt\_level="", \$encoding="url")

#### Input parameters

- \$string ... string to encode [string]
- \$key ... key of length 16 or 24 or 32 [string] (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

## 7.5.45 hcms\_decrypt

#### Description

Decryption of a string. Only strong encryption is binary-safe!

### **Syntax**

hcms\_decrypt (\$string, \$key="", \$crypt\_level="", \$encoding="url")

### Input parameters

- \$string ... hash-string to decode [string]
- \$key ... key of length 16 or 24 or 32 [string] (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

## 7.5.46 createtimetoken

### **Syntax**

createtimetoken (\$lifetime=0, \$secret=4)

#### Input parameters

- \$lifetime ... token lifetime in seconds [integer] (optional)
- \$secret ... secret value [integer] (optional)

### global input parameters

• \$mgmt\_config

#### **Output**

token / false on error

## 7.5.47 checktimetoken

### **Syntax**

checktimetoken (\$token, \$secret=4)

#### Input parameters

- \$token ... token [string]
- \$secret ... secret value [integer] (optional)

#### global input parameters

• \$mgmt\_config

# Output

true / false

### 7.5.48 createtoken

## **Syntax**

createtoken (\$user="sys", \$lifetime=0, \$secret=4)

## Input parameters

- \$user ... user name [string] (optional)
- \$lifetime ... token lifetime in seconds [integer] (optional)
- \$secret ... secret value [integer] (optional)

### global input parameters

• \$mgmt\_config

### Output

• token / false on error

### 7.5.49 checktoken

#### **Syntax**

checktoken (\$token, \$user="sys", \$secret=4)

# Input parameters

- \$token ... token [string]
- \$user ... user name [string] (optional)
- \$secret ... secret value [integer] (optional)

# global input parameters

• \$mgmt\_config

### Output

• true / false

# 7.5.50 createuniquetoken

### **Syntax**

createuniquetoken (\$length=16)

### Input parameters

• \$length ... token length [integer] (optional)

### global input parameters

• \$mgmt\_config

## Output

• token as string / false

# 7.5.51 createpassword

## **Syntax**

createpassword (\$length=10)

### Input parameters

• \$length ... password length [integer] (optional)

## global input parameters

• \$mgmt\_config

## Output

• password as string / false

### 7.5.52 rand\_secure

### Syntax

rand\_secure (\$min=1000, \$max=9999999999)

## Input parameters

- \$min ... min and max value [integer] (optional)
- \$max

### Output

secure random number / false

# 7.6 Media API Functions

### 7.6.1 ocr\_extractcontent

#### **Description**

This function extracts the text content of multimedia objects using OCR and returns the text. It is a helper function for function indexcontent.

### **Syntax**

ocr\_extractcontent (\$site, \$location, \$file)

### Input parameters

- \$site ... publication name [string]
- \$location ... path to multimedia file [string]
- \$file ... multimedia file name (file to be indexed) [string]

### global input parameters

- \$mgmt\_config
- \$mgmt\_parser
- \$mgmt\_imagepreview
- \$hcms\_lang
- \$lang

### Output

extracted content as text string / false

#### 7.6.2 indexcontent

#### 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.

#### Syntax

indexcontent (\$site, \$location, \$file, \$container="", \$container\_content="", \$user, \$return\_content=false)

- \$site ... publication name [string]
- \$location ... path to multimedia file [string]
- \$file ... multimedia file name (file to be indexed) [string]

- \$container ... container name or ID [string] (optional)
- \$container\_content ... container XML-content [string] (optional)
- \$user ... user name [string]
- \$return\_content ... return the content without saving it in the system [boolean] (optonal)

### global input parameters

- \$mgmt\_config
- \$mgmt\_parser
- \$mgmt\_imagepreview
- \$mgmt\_uncompress
- \$hcms\_ext
- \$hcms\_lang
- \$lang

#### Output

true / false

### 7.6.3 unindexcontent

## Description

This function removes media objects from the container

#### **Syntax**

unindexcontent (\$site, \$location, \$file, \$container, \$container\_content, \$user)

#### Input parameters

- \$site ... publication name [string]
- \$location ... file location [string]
- \$file ... file name [string]
- \$container ... multimedia file to index [string]
- \$container\_content ... container name or ID [string]
- \$user ... container XML-content [string]

### global input parameters

- \$mgmt\_config
- \$mgmt\_parser
- \$hcms\_lang
- \$lang

### Output

• true / false

### 7.6.4 reindexcontent

### **Description**

Reindexes all media files of a publication. Optionally only for specific containers.

#### **Syntax**

reindexcontent (\$site, \$container\_id\_array="")

#### Input parameters

- \$site ... publication name [string]
- \$container\_id\_array ... container IDs [array] (optional)

### global input parameters

• \$mgmt\_config

## Output

• true / false

## 7.6.5 createthumbnail\_indesign

### 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.

#### **Syntax**

createthumbnail\_indesign (\$site, \$location\_source, \$location\_dest, \$file)

#### Input parameters

- \$site ... publication name [string]
- \$location\_source ... path to source dir [string]
- \$location\_dest ... path to destination dir [string]
- \$file ... file name [string]

### 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)

### 7.6.6 createthumbnail\_video

### **Description**

Creates a thumbnail picture of a video frame. Saves only the thumbnail media file in destination location. Only jpeg format is supported as output.

#### **Syntax**

createthumbnail\_video (\$site, \$location\_source, \$location\_dest, \$file, \$frame, \$autorotate=false)

### Input parameters

- \$site ... publication name [string]
- \$location\_source ... path to source dir [string]
- \$location\_dest ... path to destination dir [string]
- \$file ... file name [string]
- \$frame ... frame of video in seconds or hh:mm:ss[.xxx]
- \$autorotate ... autorotate [boolean] (optional)

### global input parameters

- \$mgmt\_config
- \$mgmt\_mediapreview
- \$mgmt\_mediaoptions
- \$user

### Output

• new file name / false on error

## 7.6.7 createimages\_video

#### **Description**

Creates and saves images of video screen size from a video to a directory.

#### **Syntax**

createimages\_video (\$site, \$location\_source, \$location\_dest, \$file, \$name="", \$fs=1, \$format="jpg", \$autorotate=false, \$width="", \$height="")

- \$site ... publication name [string]
- \$location\_source ... path to source dir [string]
- \$location\_dest ... path to destination dir [string]
- \$file ... file name [string]
- \$name ... name for image files [string] (optional)
- \$fs ... frames per second to create from the video [number] (optional)
- \$format ... image format [jpg, png, bmp] (optional)
- \$autorotate ... autorotate [boolean] (optional)
- \$width ... image width in pixel [integer] (optional)
- \$height ... image height in pixel [integer] (optional)

## global input parameters

- \$mgmt\_config
- \$mgmt\_mediapreview
- \$mgmt\_mediaoptions
- \$user

#### **Output**

• true / false on error

#### 7.6.8 createmedia

### Description

Creates an new image or video from the original file 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.

### **Syntax**

createmedia (\$site, \$location\_source, \$location\_dest, \$file, \$format="", \$type="thumbnail", \$force\_no\_encrypt=false)

#### Input parameters

- \$site ... publication name [string]
- \$location\_source ... path to source dir [string]
- \$location\_dest ... path to destination dir [string]
- \$file ... file name [string]
- \$format ... format (file extension w/o dot) [string] (optional)
- \$type ... type of image/video/audio file [thumbnail(for thumbnails of images), origthumb(thumbnail made from original video/audio), original(to overwrite original video/audio file), annotation(for annotation images), any other string present in \$mgmt\_imageoptions/\$mgmt\_mediaoptions, temp(for temporary files)] (optional)
- \$force\_no\_encrypt ... force the file to be not encrypted even if the content of the publication must be encrypted [boolean] (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

# 7.6.9 splitmedia

#### Description

Splits a video or audio file in segments measured in seconds. Used for synchronous Google Cloud Speech Service that only supports max. 1 minute audio files.

#### Syntax

splitmedia (\$site, \$location\_source, \$location\_dest, \$file, \$sec=60, \$format="",
\$force\_no\_encrypt=false)

### Input parameters

- \$site ... publication name [string]
- \$location\_source ... path to source dir [string]
- \$location\_dest ... path to destination dir [string]
- \$file ... file name [string]
- \$sec ... seconds of a segment [integer] (optional)
- \$format ... target format (file extension w/o dot) of destination file [string] (optional)
- \$force\_no\_encrypt ... force the file to be not encrypted even if the content of the publication must be encrypted [boolean] (optional)

### global input parameters

- \$mgmt\_config
- \$mgmt\_imagepreview
- \$mgmt\_mediapreview
- \$mgmt\_mediaoptions
- \$mgmt\_imageoptions
- \$mgmt\_maxsizepreview
- \$mgmt\_mediametadata
- \$hcms\_ext
- \$user

#### Output

• array of new file names / false on error

### 7.6.10 convertmedia

#### Description

Converts and creates a new image/video/audio or document from original. This is a wrapper function for createmedia, createimages\_video and createdocument.

#### **Syntax**

convertmedia (\$site, \$location\_source, \$location\_dest, \$mediafile, \$format, \$media\_config="", \$force\_no\_encrypt=false)

- \$site ... publication name [string]
- \$location\_source ... path to source dir [string]
- \$location\_dest ... path to destination dir [string]
- \$mediafile ... file name [string]
- \$format ... target format (file extension w/o dot) of destination file [string]
- \$media\_config ... media configuration to be used [string] (optional)
- \$force\_no\_encrypt ... force the file to be not encrypted even if the content of the publication must be encrypted [boolean] (optional)

#### global input parameters

- \$mgmt\_config
- \$mgmt\_imagepreview
- \$mgmt\_mediapreview
- \$mgmt\_mediaoptions
- \$mgmt\_imageoptions
- \$mgmt\_maxsizepreview
- \$mgmt\_mediametadata
- \$mgmt\_compress
- \$hcms ext

# Output

• new file name / false on error

# 7.6.11 convertimage

## 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.

### **Syntax**

convertimage (\$site, \$file\_source, \$location\_dest, \$format="jpg", \$colorspace="RGB", \$iccprofile="", \$width="", \$height="", \$slug=0, \$units="px", \$dpi=72, \$quality="")

- \$site ... publication name [string]
- \$file source ... path to source image file [string]
- \$location\_dest ... path to destination dir [string]
- \$format ... format (file extension w/o dot) of destination file [string] (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 [integer] (optional)
- \$width ... height in pixel/mm/inch [integer] (optional)
- \$height ... slug in pixel/mm/inch [integer] (optional)
- \$slug ... units for width [string]
- \$units ... height and slug [px, mm, inch] (optional)
- \$dpi ... dpi [integer] (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

# 7.6.12 rotateimage

#### Description

Rotates an image (must be jpg, png or gif) using GD library. not used if ImageMagick is available

## **Syntax**

rotateimage (\$site, \$filepath, \$angle, \$imageformat)

# Input parameters

- \$site ... publication name [string]
- \$filepath ... path to source media file [string]
- \$angle ... rotation angle [integer]
- \$imageformat ... destination image format [jpg, png, gif]

#### global input parameters

- \$mgmt\_config
- \$user

## Output

• new image file name / false on error

## 7.6.13 hex2rgb

# Syntax

hex2rgb (\$hex)

• \$hex ... image color as hex-code [string]

## Output

• RGB-color as array / false on error

## 7.6.14 rgb2hex

## **Syntax**

rgb2hex (\$red, \$green, \$blue)

#### Input parameters

- \$red ... image color in RGB [string]
- \$green
- \$blue

## Output

hex-color as string / false on error

# 7.6.15 readmediaplayer\_config

### **Syntax**

readmediaplayer\_config (\$location, \$configfile)

#### Input parameters

- \$location ... path to media config file [string]
- \$configfile ... config file name [string]

### global input parameters

- \$mgmt\_config
- \$user

### Output

• config array / false on error

## 7.6.16 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 [string]
- \$configfile ... media config file name [string]
- \$mediafiles ... media file name [array or string]
- \$width ... width in px [integer] (optional)
- \$height ... height in px [integer] (optional)
- \$rotation ... rotation in degree [integer] (optional)
- \$filesize ... file size in kB [integer] (optional)
- \$duration ... duration [hh:mmm:ss] (optional)
- \$videobitrate ... video bitrate in kb/s [string] (optional)
- \$audiobitrate ... audio bitrate in kb/s [string] (optional)
- \$audiofrequenzy ... audio frequenzy in Hz [string] (optional)
- \$audiochannels ... audio channels [mono, stereo] (optional)
- \$video\_codec ... video codec name [string] (optional)
- \$audio\_codec ... audio codec name [string] (optional)

### global input parameters

- \$mgmt\_config
- \$user

### Output

• true / false on error

# 7.6.17 createdocument

#### Description

Creates a new multimedia file of given format at source destination using UNOCONV and saves it as a thumbnail file in the destination location

#### **Syntax**

createdocument (\$site, \$location\_source, \$location\_dest, \$file, \$format="",
\$force\_no\_encrypt=false)

#### Input parameters

- \$site ... publication name [string]
- \$location\_source ... path to source location [string]
- \$location\_dest ... path to destination location [string]
- \$file ... file name [string]
- \$format ... destination file format (extension w/o dot) [string]
- \$force\_no\_encrypt ... force the file to be not encrypted even if the content of the publication must be encrypted [boolean] (optional)

#### global input parameters

- \$mgmt\_config
- \$mgmt docpreview
- \$mgmt\_docoptions

- \$mgmt\_docconvert
- \$mgmt\_maxsizepreview
- \$hcms\_ext
- \$hcms\_lang
- \$lang
- \$user

# Output

new file name / false on error

# 7.6.18 unzipfile

### 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).

### **Syntax**

unzipfile (\$site, \$zipfilepath, \$location, \$filename, \$cat="comp", \$user)

### Input parameters

- \$site ... publication name [string]
- \$zipfilepath ... path to source zip file [string]
- \$location ... path to destination location [string]
- \$filename ... category [page, comp]
- \$cat ... name of file for extraction [string]
- \$user ... user name [string]

## global input parameters

- \$mgmt\_config
- \$mgmt\_uncompress
- \$mgmt\_imagepreview
- \$mgmt\_mediapreview
- \$mgmt\_mediaoptions

#### Output

result array with all object paths / false

## 7.6.19 clonefolder

### **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.

#### **Syntax**

clonefolder (\$site, \$source, \$destination, \$user, \$activity="")

### Input parameters

- \$site ... publication name [string]
- \$source ... source location [string]
- \$destination ... destination location [string]
- \$user ... user name [string]
- \$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

· container IDs as array / false

# 7.6.20 zipfiles\_helper

### Description

Compresses all files and includes their folder structure in a ZIP file. This function does not support multimedia objects and is only a helper function for native file system operations.

#### Syntax

zipfiles\_helper (\$source, \$destination, \$zipfilename, \$remove=false)

### Input parameters

- \$source ... source directory [string]
- \$destination ... destination directory [string]
- \$zipfilename ... name of ZIP-file [string]
- \$remove ... remouse all files from source [boolean] (optional)

#### global input parameters

- \$mgmt\_config
- \$mgmt\_compress

# Output

true/false

## 7.6.21 zipfiles

#### Description

Compresses all media files and includes their folder structure in a ZIP file.

#### **Syntax**

zipfiles (\$site, \$multiobject\_array, \$destination="", \$zipfilename, \$user, \$activity="",
\$flatzip=false)

#### Input parameters

- \$site ... publication name [string]
- \$multiobject\_array ... array with path to source files [array]
- \$destination ... destination location (if this is null then the \$location where the zip-file resists will be used) [string]
- \$zipfilename ... name of ZIP-file [string]
- \$user ... user name [string]
- \$activity ... activity that need to be set for daily stats [download] (optional)
- \$flatzip ... flat hierarchy means no directories [boolean] (optional)

## global input parameters

- \$mgmt\_config
- \$mgmt\_compress
- \$pageaccess
- \$compaccess
- \$hiddenfolder
- \$hcms\_linking
- \$globalpermission\$setlocalpermission
- \$hcms\_lang
- \$lang

## Output

true/false

### 7.6.22 px2mm

#### Description

Convert mm to pixel

#### **Syntax**

px2mm (\$pixel, \$dpi=72)

#### Input parameters

- \$pixel ... pixel [integer]
- \$dpi ... dpi [integer] (optional)

### Output

• pixel / false

## 7.6.23 px2inch

### Description

Convert pixel to inches

#### **Syntax**

px2inch (\$pixel, \$dpi=72)

## Input parameters

- \$pixel ... pixel [integer]
- \$dpi ... dpi [integer] (optional)

## Output

• inch / false

## 7.6.24 inch2px

### **Description**

Convert inches to pixel

# **Syntax**

inch2px (\$inch, \$dpi=72)

## Input parameters

- \$inch ... pixel [integer]
- \$dpi ... dpi [integer] (optional)

### Output

• pixel / false

# 7.6.25 sec2time

### Description

Convert seconds to time format hh: mm: ss.mmm

#### **Syntax**

sec2time (\$input)

## Input parameters

• \$input ... time in seconds [float]

### Output

• time in hh: mm: ss.mmm / false

## 7.6.26 mediasize2frame

#### **Description**

Calculates the width and height of a media to fit into a given frame size.

#### **Syntax**

mediasize2frame (\$mediawidth, \$mediaheight, \$framewidth="", \$frameheight="", \$keepmaxsize=true)

### Input parameters

- \$mediawidth ... media width [integer]
- \$mediaheight ... media height [integer]
- \$framewidth ... frame width [integer] (optional)
- \$frameheight ... frame height [integer] (optional)
- \$keepmaxsize ... keep maximum media size based on original dimensions of media without stretching [boolean] (optional)

# Output

· width and height as array / false

# 7.6.27 vtt2array

### Description

Converts VTT string to array

### **Syntax**

vtt2array (\$vtt)

#### Input parameters

• \$vtt ... video text track [string]

# Output

array / false

## 7.7 Metadata API Functions

## 7.7.1 importmetadata

## 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.

#### **Syntax**

importmetadata (\$site, \$location, \$file, \$user, \$type="", \$delimiter=";", \$enclosure='"', \$charset="utf-8")

### Input parameters

- \$site ... publication name [string]
- \$location ... location [string]
- \$file ... path to CSV file [string]
- \$user ... user name [string]
- \$type ... type array or string [u, f, I, c, d, k] (optional)
- \$delimiter ... delimiter [string] (optional)
- \$enclosure ... enclosure [string] (optional)
- \$charset ... character set [string] (optional)

### global input parameters

- \$mgmt\_config
- \$eventsystem

### Output

• true / false

### 7.7.2 loadtaxonomy

### Description

Generates an array from a taxonomy definition file located in data/include/ to be used for presentation or CSV export.

#### Syntax

loadtaxonomy (\$site, \$start=1, \$perpage=100000)

## Input parameters

- \$site ... publication name [string]
- \$start ... return rows starting with row number [integer] (optional)
- \$perpage ... return number of rows [integer] (optional)

## global input parameters

• \$mgmt\_config

#### Output

true / false

## 7.7.3 savetaxonomy

#### **Description**

Generates an array from a taxonomy definition file located in data/include/ to be used for presentation or CSV export.

### **Syntax**

savetaxonomy (\$site, \$taxonomy, \$saveindex\_start, \$saveindex\_stop)

### Input parameters

- \$site ... publication name [string]
- \$taxonomy ... taxonomy with rows and languages as keys [array]
- \$saveindex\_start ... replace rows starting with row number [integer]
- \$saveindex\_stop ... replace rows ending with row number [integer]

### global input parameters

\$mgmt\_config

### **Output**

• true / false

# 7.7.4 createtaxonomy

#### **Description**

Generates an array from a taxonomy defintion file (CSV) and saves the PHP file in data/include/publication-name.taxonomy.inc.php.

Recreates the taxonomy for all objects if the taxonomy defintion has been uodated.

### **Syntax**

createtaxonomy (\$site\_name="", \$recreate=false)

#### Input parameters

- \$site\_name ... publication name [string] (optional)
- \$recreate ... recreate taxonomy file [boolean] (optional)

#### global input parameters

\$mgmt\_config

## Output

true / false

# 7.7.5 splitkeywords

### Description

Generates a keyword list from a text by splitting and transforming the comma separated string.

#### **Syntax**

splitkeywords (\$keywords, \$charset="UTF-8")

### Input parameters

- \$keywords ... comma seperated keywords [string]
- \$charset ... character set [string] (optional)

### Output

keywords as array / false on error

## 7.7.6 copymetadata

### Description

Copies all meta data from source to destination file using EXIFTOOL

#### **Syntax**

copymetadata (\$file\_source, \$file\_dest)

### Input parameters

- \$file\_source ... path to source file [string]
- \$file\_dest ... path to destination file [string]

## global input parameters

- \$mgmt\_config
- \$mgmt\_mediametadata
- \$user

# Output

• true / false

## 7.7.7 extractmetadata

### Description

Extracts all meta data from a file using EXIFTOOL

## **Syntax**

extractmetadata (\$file)

### Input parameters

• \$file ... path to image file [string]

## global input parameters

- \$user
- \$mgmt\_config
- \$mgmt\_mediametadata

## Output

• result array / false on error

## 7.7.8 xmlobject2array

#### Description

Converts an xmlobject to an array, provided by xaviered at gmail dot com

### Syntax

xmlobject2array (\$obj, \$namespace="")

#### Input parameters

- \$obj ... XML [object]
- \$namespace ... namespace [array] (optional)

# Output

• result array / false

## 7.7.9 id3\_getdata

### Description

Requires getID3 library since EXIFTOOL cannot write ID3 tags so far

### **Syntax**

id3\_getdata (\$file)

#### Input parameters

• \$file ... path to audio file [string]

## global input parameters

- \$mgmt\_config
- \$hcms\_ext

# Output

• result array / false on error

## 7.7.10 id3\_writefile

#### **Description**

Writes ID3 tags into audio file for supported file types and keeps the existing ID3 tags

### **Syntax**

id3\_writefile (\$file, \$id3, \$keep\_data=true, \$movetempfile=true)

### Input parameters

- \$file ... abs. path to audio file [string]
- \$id3 ... ID3 tag [array]
- \$keep\_data ... keep existing ID3 data of file [boolean] (optional)
- \$movetempfile ... move tempoarary file from unecrypted to encrypted [boolean] (optional)

### global input parameters

- \$user
- \$mgmt\_config
- \$mgmt\_mediametadata
- \$hcms\_ext

## Output

true / false on error

### 7.7.11 id3\_create

### Description

Defines ID3 tag array based on the media mapping of a publication.

#### **Syntax**

id3\_create (\$site, \$text)

### Input parameters

- \$site ... publication name [string]
- \$text ... text from content container [array]

#### global input parameters

\$mgmt\_config

### Output

ID3 tag array / false on error

# 7.7.12 xmp\_getdata

### **Syntax**

xmp\_getdata (\$file)

## Input parameters

• \$file ... path to image file [string]

### global input parameters

- \$user
- \$mgmt\_config
- \$hcms\_ext

# Output

• result array / false on error

## 7.7.13 xmp\_writefile

#### Description

Writes XMP tags into image file for supported file types and keeps the existing XMP tags

#### **Syntax**

xmp\_writefile (\$file, \$xmp, \$keep\_data=true, \$movetempfile=true)

## Input parameters

- \$file ... abs. path to image file [string]
- \$xmp ... XMP tag [array]
- \$keep\_data ... keep existing XMP data of file [boolean] (optional)
- \$movetempfile ... move tempoarary file from unecrypted to encrypted [boolean] (optional)

#### global input parameters

- \$user
- \$mgmt\_config
- \$mgmt\_mediametadata
- \$hcms\_ext

### Output

• true / false on error

# 7.7.14 xmp\_create

### **Description**

Defines XMP tag array based on the media mapping of a publication

#### **Syntax**

xmp\_create (\$site, \$text)

## Input parameters

- \$site ... publication name [string]
- \$text ... text from content container [array]

## global input parameters

• \$mgmt\_config

## Output

• XMP tag array / false on error

# 7.7.15 geo2decimal

#### **Syntax**

geo2decimal (\$coordinate, \$hemisphere)

### Input parameters

- \$coordinate ... GPS location [degrees, minutes, seconds]
- \$hemisphere ... hemisphere [N, O, S, W]

## Output

decimal result / false

## 7.7.16 exif\_getdata

### **Syntax**

exif\_getdata (\$file)

### Input parameters

• \$file ... path to image file [string]

#### global input parameters

- \$user
- \$mgmt\_config
- \$hcms\_ext

### Output

result array / false

# 7.7.17 iptc\_getdata

### **Syntax**

iptc\_getdata (\$file)

#### Input parameters

• \$file ... path to image file [string]

#### global input parameters

- \$user
- \$mgmt\_config
- \$hcms\_ext

#### **Output**

result array / false

## 7.7.18 iptc\_getcharset

# 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.

### **Syntax**

iptc\_getcharset (\$tag)

# Input parameters

• \$tag ... iptc tag that holds character set information [string]

### Output

character set as string / false on error

## 7.7.19 iptc\_maketag

### Description

Convert the IPTC tag to binary code

#### **Syntax**

iptc\_maketag (\$record=2, \$tag, \$value)

#### Input parameters

- \$record ... type of tag (e.g. 2) [integer]
- \$tag ... code of tag (e.g. 025) [string]
- \$value ... value of tag [string]

## Output

• binary IPTC tag / false on error

### 7.7.20 iptc\_writefile

## Description

Writes IPTC tags into image file for supported file types and keeps the existing IPTC tags

#### **Syntax**

iptc\_writefile (\$file, \$iptc, \$keep\_data=true, \$movetempfile=true)

#### Input parameters

- \$file ... abs. path to image file [string]
- \$iptc ... IPTC tag [array]
- \$keep\_data ... keep existing IPTC data of file [boolean] (optional)
- \$movetempfile ... move tempoarary file from unecrypted to encrypted [boolean] (optional)

## global input parameters

- \$user
- \$mgmt\_config

• \$mgmt\_mediametadata

# Output

true / false on error

# 7.7.21 iptc\_create

### Description

Defines IPTC tag array based on the media mapping of a publication

#### **Syntax**

iptc\_create (\$site, \$text)

## Input parameters

- \$site ... publication name [string]
- \$text ... text from content container [array]

# global input parameters

• \$mgmt\_config

### Output

• IPTC tag array / false on error

## 7.7.22 createmapping

## Description

Prepares the PHP mapping array from the provided mapping definition and saves media mapping file

## **Syntax**

createmapping (\$site, \$mapping)

## Input parameters

- \$site ... publication name [string]
- \$mapping ... mapping definition [string]

### global input parameters

\$mgmt\_config

## Output

• true / false on error

# 7.7.23 getmapping

#### Description

Load the mapping file of the provided publication.

#### **Syntax**

getmapping (\$site)

### Input parameters

\$site ... publication name [string]

### global input parameters

\$mgmt\_config

#### **Output**

mapping code for display / false

### 7.7.24 setmetadata

### 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.

#### **Syntax**

```
setmetadata ($site, $location="", $object="", $mediafile="", $mapping="", $containerdata="", $user, $savecontainer=true)
```

### Input parameters

- \$site ... publication name [string]
- \$location ... location path [string] (optional)
- \$object ... object name [string] (optional)
- \$mediafile ... media file name [string] (optional)
- \$mapping ... mapping [array:metadata-tag-name => text-id] (optional)
- \$containerdata ... container content as XML [string] (optional)
- \$user ... user name [string]
- \$savecontainer ... save content container [boolean] (optional)

#### global input parameters

- \$eventsystem
- \$mgmt\_config
- \$hcms\_ext

### Output

container content as XML string / false

## 7.8 Link API Functions

## 7.8.1 link\_db\_restore

### Description

This function restores a given or all link management index files

### **Syntax**

link\_db\_restore (\$site="")

### Input parameters

• \$site ... publication name [string] (optional)

### global input parameters

• \$mgmt\_config

### Output

• true / false on error

### 7.8.2 link db load

#### 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.

#### **Syntax**

link\_db\_load (\$site, \$user)

#### Input parameters

- \$site ... publication name [string]
- \$user ... user name [string]

## global input parameters

• \$mgmt\_config

#### Output

• link database [2 dim. array] or true / false on error

## 7.8.3 link\_db\_read

### **Description**

This function loads the link management database for reading without locking

### **Syntax**

link\_db\_read (\$site)

## Input parameters

• \$site ... publication name [string]

## global input parameters

• \$mgmt\_config

### Output

• link database [2 dim. array] or true / false on error

## 7.8.4 link\_db\_close

### Description

closes and unlocks the link management database.

## **Syntax**

link\_db\_close (\$site, \$user)

## Input parameters

- \$site ... publication name [string]
- \$user ... user name [string]

### global input parameters

\$mgmt\_config

# Output

true/false

## 7.8.5 link\_db\_save

### Description

This function saves und unlocks the link management database

#### **Syntax**

link\_db\_save (\$site, \$link\_db, \$user)

#### Input parameters

- \$site ... link database [array]
- \$link\_db ... publication name [string]
- \$user ... user name [string]

# global input parameters

• \$mgmt\_config

#### Output

• true/false on error

### 7.8.6 link\_db\_update

### 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 link matching given category)

link\_curr & link\_new are not empty and not equal: update current link with the new one

#### **Syntax**

```
link_db_update ($site, $link_db, $attribute, $contentfile="", $cat="", $link_curr="", $link_new="", $option)
```

#### Input parameters

- \$site ... publication name [string]
- \$link\_db ... link database [2 dim. array]
- \$attribute ... attribute [object, link]
- \$contentfile ... content container [string] (optional)
- \$cat ... link category [comp, page] (optional)
- \$link\_curr ... current link must be an URL or absolute/relative path [string] (optional)
- \$link\_new ... new link must be an URL or absolute/relative path [string] (optional)
- \$option ... update option [all, unique]

## global input parameters

\$mgmt\_config

#### **Output**

link database [array] or true if link index database is not used / false on error

## 7.8.7 link\_db\_insert

## Description

This function inserts a new record in the link management database optionally the created object can be also inserted

# **Syntax**

link\_db\_insert (\$site, \$link\_db, \$contentfile, \$cat, \$object="")

### Input parameters

- \$site ... publication name [string]
- \$link\_db ... link database [2 dim. array]
- \$contentfile ... content container name [string]
- \$cat ... link category [comp, page]
- \$object ... object name (optional)

### global input parameters

• \$mgmt\_config

### Output

• link database [2 dim. array] or true / false

### 7.8.8 link\_db\_delete

#### **Description**

This function deletes a record in the link management database

#### **Syntax**

link\_db\_delete (\$site, \$link\_db, \$contentfile)

#### Input parameters

- \$site ... link database [2 dim. array]
- \$link\_db ... content container name [string]
- \$contentfile

#### global input parameters

\$mgmt\_config

## Output

• link database [2 dim. array] or true / false on error

# 7.8.9 link\_db\_getobject

### Description

This function splits the object string into an array of objects.

### **Syntax**

link\_db\_getobject (\$multiobject)

### Input parameters

• \$multiobject ... link database attribut (references to objects seperated by |) [string]

# global input parameters

• \$mgmt\_config

# Output

• objects [array] / false on error

## 7.8.10 link\_update

### Description

This function updates the link of the published and working content container and link file

#### Syntax

link\_update (\$site, \$container, \$link\_old, \$link\_new)

### Input parameters

- \$site ... publication name [string]
- \$container ... container name [string]
- \$link\_old ... old link (converted) [string]
- \$link\_new ... new link (converted) [string]

### global input parameters

- \$user
- \$mgmt\_config

# Output

true/false

# 7.8.11 getlinkedobject

### 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.

## **Syntax**

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

### Input parameters

- \$site ... publication name [string]
- \$location ... location [string]
- \$page ... object (name and extension) [string]
- \$cat ... category [page, comp] (optional)

### global input parameters

\$mgmt\_config

### Output

• objects which link to the given object [array] or true / false

# 7.8.12 getconnectedobject

#### **Description**

This function gets all objects which use the same content container and are therefore connected.

#### **Syntax**

getconnectedobject (\$container, \$type="work")

#### Input parameters

- \$container ... container name [string]
- \$type ... container type [work, published, version] (optional)

# global input parameters

- \$mgmt\_config
- \$user

## Output

connected objects[array]

### 7.8.13 extractlinks

### **Description**

This function extracts all links based on it's identifier from a text and returns an array of all

#### links

#### **Syntax**

extractlinks (\$textcontent, \$identifier)

#### Input parameters

- \$textcontent ... text content [string]
- \$identifier ... link identifiert ["href" for hyperreferences, "src" for image references or embed references (flash), "value" for paramter (flash)]

### global input parameters

• \$mgmt\_config

## Output

• object links [array] / false on error

# 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

#### 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.

#### **Syntax**

plugin\_readmenu (\$xml, \$pluginFolder)

- \$xml ... plugin xml [string]
- \$pluginFolder ... plugin directory [string]

\$mgmt\_config

## Output

menu point array used by navigator

# 7.9.3 plugin\_parse

### **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.

#### **Syntax**

plugin\_parse (\$oldData=array())

### Input parameters

• \$oldData ... mgmt\_plugin [array] (optional)

# global input parameters

• \$mgmt\_config

### Output

mgmt\_plugin as array

## 7.9.4 plugin\_generatedefinition

# 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.

#### Syntax

plugin\_generatedefinition (\$arrayName, \$array)

- \$arrayName ... name of array holding the plugin definitions [string]
- \$array ... configuration containing the values and keys [array]

\$mgmt\_config

## Output

plugin array / false on error

# 7.9.5 plugin\_saveconfig

### **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.

### **Syntax**

plugin\_saveconfig (\$configuration)

## Input parameters

• \$configuration ... configuration [array]

### global input parameters

• \$mgmt\_config

## Output

• true / false on error

# 7.9.6 plugin\_generatelink

### **Description**

Generates a link to be used when linking to other pages inside of a plugin

# **Syntax**

plugin\_generatelink (\$plugin, \$page, \$control=false, \$additionalGetParameters=false)

## Input parameters

- \$plugin ... plugin name [string]
- \$page ... plugin page (relative reference to the plugins main page) [string]
- \$control ... control (relative reference to the plugins control page) [string]
- \$additionalGetParameters ... additional GET parameters [string]

### global input parameters

• \$mgmt\_config

plugin link

# 7.10 User Interface API Functions

## 7.10.1 windowwidth

## Description

Returns the width of the object window when editing/opening an object

## **Syntax**

windowwidth (\$type="object")

## Input parameters

• \$type ... type [string] (optional)

## global input parameters

• \$mgmt\_config

## Output

• window width in pixels

# 7.10.2 windowheight

## Description

Returns the height of the object window when editing/opening an object

## **Syntax**

windowheight (\$type="object")

## Input parameters

• \$type ... type [string] (optional)

# global input parameters

• \$mgmt\_config

## Output

• window height in pixels

# 7.10.3 toggleview

## Description

Set explorer objectlist view parameter

## **Syntax**

toggleview (\$view)

# Input parameters

• \$view ... view [detail, small, medium, large]

# global input parameters

• \$mgmt\_config

# Output

• true / false

# 7.10.4 togglesidebar

# Description

Enables or disables the sidebar

## **Syntax**

togglesidebar (\$view)

## Input parameters

• \$view ... view [boolean]

# global input parameters

• \$mgmt\_config

## Output

• true / false

# 7.10.5 setfilter

# Description

Set filter settings for object view in session

## **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

# 7.10.6 objectfilter

### Description

If an object name is passing the filter-test. One or more filters need to be set in the session "hcms\_objectfilter".

### **Syntax**

objectfilter (\$file)

## Input parameters

• \$file ... file name [string]

## global input parameters

- \$mgmt\_config
- \$hcms\_ext

## Output

• true / false

### 7.10.7 showdate

### **Description**

Prepares the date and time for the display in the users time zone and format.

#### Syntax

showdate (\$date, \$sourceformat="Y-m-d H:i", \$targetformat="Y-m-d H:i", \$timezone=true)

# Input parameters

• \$date ... date and time [string

- \$sourceformat ... date input format [string]
- \$targetformat ... date output format [string]
- \$timezone ... correct time zone [boolean] (optional)

• \$mgmt\_config

#### **Output**

date and time

## 7.10.8 showshorttext

### Description

Reduce the length of a string and add "..." at the end

#### **Syntax**

showshorttext (\$text, \$length=0, \$linebreak=false, \$charset="UTF-8")

### Input parameters

- \$text ... text [string]
- \$length ... max. length of text (minus length starting from the end) [integer] (optional)
- \$linebreak ... line break instead of cut [boolean] only if length is positive (optional)
- \$charset ... character set for encoding [string] (optional)

## Output

- shortened text if possible
- or orignal text

## 7.10.9 showtopbar

### Description

Returns the standard top bar with or without close button

#### **Syntax**

showtopbar (\$show, \$lang="en", \$close\_link="", \$close\_target="", \$individual\_button="", \$id="bar")

- \$show ... message [string]
- \$lang ... language code [string] (optional)
- \$close\_link ... close button link [string] (optional)
- \$close\_target ... link target [string] (optional)
- \$individual\_button ... individual button [string] (optional)

• \$id ... ID of div-layer [string] (optional)

## global input parameters

- \$mgmt\_config
- \$hcms\_charset
- \$hcms\_lang

# Output

• top bar box / false on error

# 7.10.10 showtopmenubar

### **Description**

Returns the menu top bar with or without close button

### **Syntax**

showtopmenubar (\$show, \$menu\_array, \$lang="en", \$close\_link="", \$close\_target="", \$id="bar")

### Input parameters

- \$show ... message [string]
- \$menu\_array ... menu [array:name => properties/events]
- \$lang ... language code [string] (optional)
- \$close\_link ... close button link [string] (optional)
- \$close\_target ... link target [string] (optional)
- \$id ... ID of div-layer [string] (optional)

### global input parameters

- \$mgmt\_config
- \$hcms\_charset
- \$hcms\_lang

# Output

• top bar box / false on error

## 7.10.11 showhomeboxes

## Description

Returns the file path to the home boxes

## **Syntax**

showhomeboxes (\$homebox\_array)

## Input parameters

• \$homebox\_array ... home box names [array: file name => readable name]

## global input parameters

• \$mgmt\_config

# Output

path to home boxes as array / false on error

## 7.10.12 showmessage

### **Description**

Returns the standard message box with close button

### **Syntax**

showmessage (\$show, \$width="580px", \$height="70px", \$lang="en", \$style="", \$id="hcms\_messageLayer")

### Input parameters

- \$show ... message [string]
- \$width ... width in pixel [integer] (optional)
- \$height ... height in pixel [integer] (optional)
- \$lang ... language code [string] (optional)
- \$style ... additional style definitions of div-layer [string] (optional)
- \$id ... ID of div-layer [string] (optional)

## global input parameters

- \$mgmt\_config
- \$hcms\_charset
- \$hcms\_lang

## Output

message box / false on error

# 7.10.13 showinfopage

### **Description**

Returns a full html info page

#### **Syntax**

showinfopage (\$show, \$lang="en", \$onload="")

## Input parameters

- \$show ... message [string]
- \$lang ... language code [string] (optional)
- \$onload ... on load JS events [string] (optional)

### global input parameters

- \$mgmt\_config
- \$hcms\_charset
- \$hcms\_lang\_codepage
- \$hcms\_lang

## Output

• message on html info page / false on error

## 7.10.14 showinfobox

#### **Description**

Returns the infobox as long as it has not been closed. Saves the close event in localstorage of browser.

#### **Syntax**

showinfobox (\$show, \$lang="en", \$style="", \$id="hcms\_infoboxLayer")

### Input parameters

- \$show ... message [string]
- \$lang ... language code [string] (optional)
- \$style ... additional style definitions of div-layer [string] (optional)
- \$id ... ID of div-layer [string] (optional)

## global input parameters

- \$mgmt\_config
- \$hcms\_charset
- \$hcms\_lang\_codepage
- \$hcms\_lang

### Output

• message in div layer / false on error

# 7.10.15 showsharelinks

## **Description**

Returns the presenation of share links of social media platforms

#### **Syntax**

showsharelinks (\$link, \$mediafile, \$lang="en", \$style="", \$id="hcms\_shareLayer")

### Input parameters

- \$link ... link to share [string]
- \$mediafile ... media file name [string]
- \$lang ... language code [string] (optional)
- \$style ... additional style definitions of div-layer [string] (optional)
- \$id ... ID of div-layer [string] (optional)

## global input parameters

- \$mgmt\_config
- \$hcms\_charset
- \$hcms\_lang\_codepage
- \$hcms\_lang

## Output

• message in div layer / false on error

## 7.10.16 showmetadata

#### **Syntax**

showmetadata (\$data, \$lang="en", \$class\_headline="hcmsRowData2")

### Input parameters

- \$data ... metadata [array]
- \$lang ... 2 digits language code [string]
- \$class\_headline ... CSS-class with background-color for headlines [string] (optional)

## global input parameters

- \$mgmt\_config
- \$hcms\_charset
- \$hcms\_lang\_codepage
- \$hcms\_lang

## Output

result as HTML unordered list / false on error

# 7.10.17 showobject

### **Syntax**

showobject (\$site, \$location, \$page, \$cat="", \$name="")

### Input parameters

- \$site ... publication name [string]
- \$location ... location [string]
- \$page ... object name [string]
- \$cat ... category [page, comp] (optional)
- \$name ... object name [string] (optional)

### global input parameters

- \$mgmt\_config
- \$hcms\_charset
- \$hcms\_lang
- \$hcms\_lang\_date
- \$lang

## Output

• html presentation / false

#### 7.10.18 showmedia

## Description

This function requires site, location and cat to be set as global variable in order to validate the access permission of the user

### **Syntax**

showmedia (\$mediafile, \$medianame, \$viewtype, \$id="", \$width="", \$height="", \$class="hcmsImageItem")

### Input parameters

- \$mediafile ... mediafile (publication/filename) [string]
- \$medianame ... name of mediafile for display [string]
- \$viewtype ... view type [template, media\_only, preview, preview\_download, preview\_no\_rendering]
- \$id ... ID of the HTML media tag [string]
- \$width ... width in px [integer] (optional)
- \$height ... height in px [integer] (optional)
- \$class ... CSS class [string] (optional)

## global input parameters

- \$site
- \$mgmt\_config
- \$mgmt\_mediapreview
- \$mgmt\_mediaoptions

- \$mgmt\_imagepreview
- \$mgmt\_docpreview
- \$mgmt\_docoptions
- \$mgmt\_docconvert
- \$mgmt\_maxsizepreview
- \$hcms\_charset
- \$hcms\_lang\_codepage
- \$hcms\_lang\_date
- \$hcms\_lang
- \$lang
- \$site
- \$location
- \$cat
- \$page
- \$user
- \$pageaccess
- \$compaccess
- \$downloadformats
- \$hiddenfolder
- \$hcms\_linking
- \$setlocalpermission
- \$mgmt\_imageoptions
- \$is\_mobile
- \$is\_iphone

html presentation of any media asset / false

# 7.10.19 showcompexplorer

#### Description

Creates component explorer including the search form

#### Syntax

```
showcompexplorer ($site, $dir, $location_esc="", $page="", $compcat="multi", $search_expression="", $search_format="", $mediatype="", $lang="en", $callback="", $scalingfactor="1", $view="list", $thumbsize=100)
```

- \$site ... publication name [string]
- \$dir ... current explorer location [string]
- \$location\_esc ... object location [string] (optional)
- \$page ... object name [string] (optional)
- \$compcat ... component category [single, multi, media] (optional)
- \$search\_expression ... search expression [string] (optional)
- \$search\_format ... search format [object, document, image, video, audio, watermark] (optional)
- \$mediatype ... media-type [audio, video, text, flash, image, compressed, binary] (optional)
- \$lang ... view tpye [list, gallery] (optional)
- \$callback ... thumbnail size in pixel [integer] callback of CKEditor [string] (optional)
- \$scalingfactor ... saclingfactor for images [integer] (optional)
- \$view

• \$thumbsize

### 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

### 7.10.20 showeditor

### Description

Returns the rich text editor code

# **Syntax**

showeditor (\$site, \$hypertagname, \$id, \$contentbot="", \$sizewidth=600, \$sizeheight=300, \$toolbar="Default", \$lang="en", \$dpi=72)

### Input parameters

- \$site ... publication name [string]
- \$hypertagname ... hypertag name [string]
- \$id ... hypertag id [string]
- \$contentbot ... content [string]
- \$sizewidth ... width of the editor [integer]
- \$sizeheight ... height of the editor [integer]
- \$toolbar ... toolbar set [string]
- \$lang ... 2 digit language code [string]
- \$dpi ... dpi for scaling images [integer]

# global input parameters

- \$mgmt\_config
- \$publ\_config

# Output

rich text editor code / false on error

# 7.10.21 showinlineeditor\_head

### **Description**

Returns the rich text editor code (JS, CSS) for include into the html head section

### **Syntax**

showinlineeditor\_head (\$lang)

## Input parameters

• \$lang ... 2 digit language code [string]

## global input parameters

- \$mgmt\_config
- \$hcms\_charset
- \$hcms\_lang

### Output

• rich text editor code for html head section / false on error

# 7.10.22 showinlinedatepicker\_head

## Description

Returns the date picker code (JS, CSS) for include into the html head section

#### **Syntax**

showinlinedatepicker\_head ()

### Input parameters

# global input parameters

\$mgmt\_config

## Output

• date picker code for html head section / false on error

### 7.10.23 showinlineeditor

### Description

shows the rich text inline editor

#### **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 [string]
- \$hypertag ... hypertag [string]
- \$id ... hypertag id [string]
- \$contentbot ... content [string]
- \$sizewidth ... width of the editor [integer]
- \$sizeheight ... height of the editor [integer]
- \$toolbar ... toolbar set [string]
- \$lang ... 2 digit language code [string]
- \$contenttype ... content-type [string]
- \$cat ... category [page, comp]
- \$location\_esc ... converted location [string]
- \$page ... object name [string]
- \$contentfile ... container name [string]
- \$db\_connect ... DB-connect file name [string]
- \$token ... security token [string]

### global input parameters

- \$mgmt\_config
- \$publ\_config
- \$hcms\_charset
- \$hcms\_lang

## Output

rich text editor code / false on error

## 7.10.24 showvideoplayer

#### Description

Generates a html segment for the video player code

### **Syntax**

showvideoplayer (\$site, \$video\_array, \$width=854, \$height=480, \$logo\_url="", \$id="", \$title="", \$autoplay=true, \$fullscreen=true, \$loop=false, \$muted=false, \$controls=true, \$iframe=false, \$force\_reload=false)

- \$site ... video array containing the different html sources [array]
- \$video\_array ... width of the video in pixel [integer]
- \$width ... height of the video in pixel [integer]
- \$height ... link to the logo which is displayed before you click on play (If the value is null the default logo will be used) [string]
- \$logo\_url ... ID of the video (will be generated when empty) [string]
- \$id ... title for this video [string]
- \$title ... autoplay video on load (true)
- \$autoplay ... default is false [boolean]

- \$fullscreen ... view the video in full screen [boolean]
- \$loop ... play loop [boolean] (optional)
- \$muted ... muted/no sound [boolean] (optional)
- \$controls ... player controls [boolean] (optional)
- \$iframe ... use video in iframe [boolean] (optional)
- \$force\_reload ... reload video sources to prevent the browser cache to show the same video even if it has been changed [boolean] (optional)

• \$mgmt\_config

## Output

• HTML code of the video player / false on error

## 7.10.25 showvideoplayer\_head

#### **Syntax**

showvideoplayer\_head (\$secureHref=true, \$fullscreen=true)

### Input parameters

- \$secureHref ... secure hyperreferences by adding 'hypercms\_' [boolean] (optional)
- \$fullscreen ... is it possible to view the video in fullScreen [boolean] (optional)

## global input parameters

\$mgmt\_config

### **Output**

• head for video player / false on error

## 7.10.26 showaudioplayer

# Description

Generates the html segment for the video player code

## **Syntax**

showaudioplayer (\$site, \$audioArray, \$width=320, \$height=320, \$logo\_url="", \$id="", \$autoplay=false, \$loop=false, \$controls=true, \$force\_reload=false)

- \$site ... publication name [string]
- \$audioArray ... audio files [array]
- \$width ... ID of the tag [string] (optional)

- \$height ... width of the video in pixel [integer]
- \$logo\_url ... height of the video in pixel [integer]
- \$id ... link to the logo which is displayed before you click on play (If the value is null the default logo will be used) [string]
- \$autoplay ... ID of the video (will be generated when empty) [string]
- \$loop ... autoplay (optional) [boolean]
- \$controls ... play loop (optional) [boolean]
- \$force\_reload ... player controls (optional) [boolean]

• \$mgmt\_config

## Output

• code of the HTML5 player / false

# 7.10.27 showaudioplayer\_head

#### **Syntax**

showaudioplayer\_head (\$secureHref=true)

### Input parameters

• \$secureHref ... secure hyperreferences by adding 'hypercms\_' [boolean] (optional)

# global input parameters

• \$mgmt\_config

### **Output**

head for audio player

# 7.10.28 debug\_getbacktracestring

#### Description

Returns the current backtrace as a good readable string. Ignores debug and debug\_getbacktracestring.

#### **Syntax**

debug\_getbacktracestring (\$valueSeparator, \$rowSeparator, \$ignoreFunctions=array())

- \$valueSeparator ... separator for arguments [string]
- \$rowSeparator ... separator for a row on screen/file [string]
- \$ignoreFunctions ... functionnames to be ignored [array]

debug message

#### 7.10.29 showAPIdocs

### **Description**

Generates the documentation of an API file

### **Syntax**

showAPIdocs (\$file, \$return="html", \$html\_hr=true, \$html\_description=true, \$html\_input=true, \$html\_globals=true, \$html\_output=true)

### Input parameters

- \$file ... path to API file [string]
- \$return ... return result as HTML or array [html, array] (optional)
- \$html\_hr ... use horizontal rule as separator in HTML output [boolean] (optional) display description [boolean] (optional)
- \$html\_description ... display input parameters [boolean] (optional)
- \$html\_input ... display global variables [boolean] (optional)
- \$html\_globals ... display output [boolean] (optional)
- \$html\_output

#### global input parameters

= array()

#### Output

• HTML output of documentation / false on error

# 7.10.30 readnavigation

### Description

Reads the content from the container and collects information about a single navigation item

### **Syntax**

readnavigation (\$site, \$docroot, \$object, \$view="publish", \$user="sys")

- \$site ... publication name [string]
- \$docroot ... location [string]
- \$object ... object name [string]
- \$view ... view name (see view parameters of function buildview) [string]
- \$user ... user name [string]

- \$mgmt\_config
- \$navi\_config

### Output

navigation item array / false

## 7.10.31 createnavigation

```
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-
togale=\"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);
```

#### Syntax

```
createnavigation ($site, $docroot, $urlroot, $view="publish", $currentobject="",
$recursive=true)
```

- \$site ... publication name [string]
- \$docroot ... document root for navigation [string]
- \$urlroot ... URL root for navigation [string]
- \$view ... view name (see view parameters of function buildview) [string]
- \$currentobject ... path to current object [string] (optional)
- \$recursive ... recursive [boolean] (optional)

- \$mgmt\_config
- \$navi\_config

## Output

navigation array / false

## 7.10.32 shownavigation

## 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-
togale=\"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 [boolean]
$navi_config['use_1st_folderitem'] = false;
```

### **Syntax**

shownavigation (\$navigation, \$level=1)

### Input parameters

- \$navigation ... navigation (created by function readnavigation) [array]
- \$level ... level [integer] (optional)

### global input parameters

- \$mgmt\_config
- \$navi\_config

### **Output**

navigation HTML presentation / false

### 7.10.33 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) [array]
- \$selected\_value ... use values of array as option value and text [boolean] (optional)
- \$id ... selected value [string] (optional)
- \$attributes ... attributes of select tags like name or id or events [string] (optional)

### Output

• HTML select box presentation / false

### 7.10.34 showtranslator

### **Syntax**

showtranslator (\$site, \$id, \$type, \$charset="UTF-8", \$lang="en", \$style="")

#### Input parameters

- \$site ... publication name [string]
- \$id ... editor/text-tag ID [string]
- \$type ... unformatted or formatted texttag-type [u, f]
- \$charset ... character set [string] (optional)
- \$lang ... 2 digit language code [string] (optional)
- \$style ... style of div tag [string] (optional)

### global input parameters

- \$mgmt\_config
- \$hcms\_lang

• HTML translator box presentation / false

# 7.10.35 showmapping

#### **Description**

Present the mapping form of the provided publication.

#### Syntax

showmapping (\$site, \$lang="en")

# Input parameters

- \$site ... publication name [string]
- \$lang ... 2 digit language code [string] (optional)

# global input parameters

- \$mgmt\_config
- \$hcms\_charset
- \$hcms\_lang

## Output

• table with form fields for display / false

# 7.10.36 showgallery

# Description

Presents all objects in a gallery with their thumbnails.

#### **Syntax**

showgallery (\$multiobject, \$thumbsize=100, \$openlink=false, \$user="sys")

### Input parameters

- \$multiobject ... multiobjects represented by their path or object ID [array]
- \$thumbsize ... thumbnail size in pixels [integer] (optional)
- \$openlink ... open object on click [boolean] (optional)
- \$user ... user name [string] (optional)

## global input parameters

• \$mgmt\_config

- \$pageaccess
- \$compaccess
- \$hiddenfolder
- \$hcms\_linking
- \$globalpermission
- \$setlocalpermission
- \$hcms\_lang
- \$lang

• gallery view / false

### 7.10.37 showthumbnail

#### Description

Presents the thumbnail of a single media file that is optionally base64 encoded an can embedded in HTML pages or e-mails.

### **Syntax**

showthumbnail (\$site, \$mediafile, \$name="", \$thumbsize=120, \$base64=false, \$style="", \$theme="standard")

## Input parameters

- \$site ... publication name [string]
- \$mediafile ... media file name [string]
- \$name ... display name [string] (optional)
- \$thumbsize ... thumbnail size in pixels [integer] (optional)
- \$base64 ... base64 encoding [boolean] (optional)
- \$style ... CSS style for image [string] (optional) design theme name for icons [string] (optional)
- \$theme

### global input parameters

• \$mgmt\_config

## Output

• thumbnail view / false

### 7.10.38 showworkflowstatus

#### Description

Displays the workflow status information table.

#### **Syntax**

showworkflowstatus (\$site, \$location, \$page)

## Input parameters

- \$site ... publication name [string]
- \$location ... location path [string]
- \$page ... object name [string]

## global input parameters

- \$mgmt\_config
- \$publ\_config
- \$hcms\_charset
- \$hcms\_lang
- \$lang

## Output

workflow status view / false

# 7.11 Template Engine API Functions

## 7.11.1 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

## 7.11.2 followlink

### **Syntax**

followlink (\$site, \$follow)

- \$site ... publication name
- \$follow ... link to follow

• \$mgmt\_config

### Output

• prepared input (location plus page) for easyedit mode (buildview) / false on error

## 7.11.3 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

### 7.11.4 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: "template or any other word": the standard text (in table) will be included for the view "preview": preview of the content of the included file "publish": view the content of the included file as it is (for publishing)
- \$charset

#### global input parameters

- \$user
- \$mgmt\_config
- \$location

- \$hcms\_lang
- \$lang

• view of the content including the content of included objects

#### 7.11.5 buildview

#### **Description**

buildview parameter may have the following values:

"formedit": use form for content editing

"formmeta": use form for content viewing only for meta informations (tag-type must be meta)

"formlock": use form for content viewing

"cmsview": view of page based on template, includes hyperCMS specific code (buttons)

"inlineview": view of page based on template, includes hyperCMS specific code (buttons) and inline text editing

"publish": view of page for publishing based on template without CMS specific code (editing)

"unpublish": execution of the code for unpublishing an object

"preview": view of page based on template for preview (inactive hyperlinks) without CMS specific code (buttons)

"template": view of template based on template for preview (inactive hyperlinks) without CMS specific code (buttons)

## **Syntax**

buildview (\$site, \$location, \$page, \$user, \$buildview="template", \$ctrlreload="no", \$template="", \$container="", \$force cat="", \$execute code=true)

## Input parameters

- \$site ... publication name [string]
- \$location ... location [string]
- \$page ... object name [string]
- \$user ... user name [string]
- \$buildview ... view parameter [string] (optional)
- \$ctrlreload ... reload workplace control frame and add html & body tags if missing [yes, no] (optional)
- \$template ... template name [string] (optional)
- \$container ... container name [string] (optional)
- \$force\_cat ... force category to use different location path [page, comp] (optional)
- \$execute\_code ... execute\_code [boolean] (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

result array with view of the content / false on error

## 7.11.6 unescapeview

## **Syntax**

unescapeview (\$viewstore, \$application="php")

### Input parameters

- \$viewstore ... code [string]
- \$application ... application name [string] (optional)

### Output

• unescaped code / false on error

## 7.11.7 buildsearchform

### **Syntax**

buildsearchform (\$site="", \$template="", \$report="", \$ownergroup="", \$css\_display="inline-block", \$css\_width\_field="90%", \$empty\_values=true, \$title="")

#### Input parameters

- \$site ... publication name [string] (optional for report)
- \$template ... template name [string] (optional)
- \$report ... or report name [string] (optional)
- \$ownergroup ... group access [array] (optional)
- \$css\_display ... CSS display value for label tag [string] (optional)
- \$css\_width\_field ... CSS field width (optional)
- \$empty\_values ... allow empty values [boolean] (optional)
- \$title ... display title [string] (optional)

## global input parameters

• \$user

- \$siteaccess
- \$mgmt\_config
- \$mgmt\_lang\_shortcut\_default
- \$hcms\_charset
- \$hcms\_lang\_name
- \$hcms\_lang\_shortcut
- \$hcms\_lang\_codepage
- \$hcms\_lang\_date
- \$hcms\_lang
- \$lang

form view / false on error

### 7.11.8 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 [string]
- \$paper\_width ... width of paper in pixel [integer]
- \$paper\_height ... height of paper in pixel [integer]
- \$paper\_top ... top space in pixel [integer]
- \$paper\_left ... left space in pixel [integer]
- \$x\_axis ... x-axis values [array]
- \$y1\_axis ... y1-axis values [array]
- \$y2\_axis ... y2-axis values [array] (optional)
- \$y3\_axis ... y3-axis values [array] (optional)
- \$paper\_style ... paper CSS style [string]
- \$bar1\_style ... 1st bar chart CSS style [string]
- \$bar2\_style ... 2nd bar chart CSS style [string]
- \$bar3\_style ... 3rd bar chart CSS style [string]
- \$show\_value ... show y-value in bar [boolean]

### global input parameters

- \$lang
- \$mgmt\_config

#### **Output**

bar chart view / false on error

## 7.12 XML API Functions

# 7.12.1 valid\_tagname

#### Description

Verifies a tag name

### **Syntax**

valid\_tagname (\$tagname)

## Input parameters

• \$tagname ... tag name [string]

### Output

• true / false on error

# 7.12.2 setxmlparameter

#### **Description**

Set parameter values in XML declaration (e.g. encoding): encoding="UTF-8"

## **Syntax**

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

## Input parameters

- \$xmldata ... XML content container [string]
- \$parameter ... paramater name [string]
- \$value ... paramater value [string]

# Output

• XML content container / false on error

## 7.12.3 getcontent

## 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.

#### Syntax

getcontent (\$xmldata, \$starttagname)

- \$xmldata ... XML content container [string]
- \$starttagname ... tag name [string]

• result array with the content of the requested XML node (tag) / false on error

## 7.12.4 geticontent

#### **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

#### Syntax

geticontent (\$xmldata, \$starttagname)

### Input parameters

- \$xmldata ... XML content container [string]
- \$starttagname ... tag name [string]

## Output

• result array with the content of the requested XML node (tag) / false on error

# 7.12.5 getxmlcontent

#### 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.

#### **Syntax**

getxmlcontent (\$xmldata, \$starttagname)

#### Input parameters

- \$xmldata ... XML content container [string]
- \$starttagname ... tag name [string]

## Output

• result array with the content of the requested XML node (tag) / false on error

# 7.12.6 getxmlicontent

### **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.

### **Syntax**

getxmlicontent (\$xmldata, \$starttagname)

## Input parameters

- \$xmldata ... XML content container [string]
- \$starttagname ... tag name [string]

## Output

• result array with the content of the requested XML node (tag) / false on error

### 7.12.7 selectcontent

# 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!

## **Syntax**

selectcontent (\$xmldata, \$starttagname, \$startcondtag, \$condvalue)

### Input parameters

- \$xmldata ... XML content container [string]
- \$starttagname ... tag name of requested XML node [string]
- \$startcondtag ... tag holding the conditional value inside the given starttagname [string]
- \$condvalue ... conditional value [string]

### **Output**

result array with the content of the requested XML node (tag) / false on error

## 7.12.8 selecticontent

### 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!

### **Syntax**

selecticontent (\$xmldata, \$starttagname, \$startcondtag, \$condvalue)

#### Input parameters

- \$xmldata ... XML content container [string]
- \$starttagname ... tag name of requested XML node [string]
- \$startcondtag ... tag holding the conditional value inside the given starttagname [string]
- \$condvalue ... conditional value [string]

## Output

result array with the content of the requested XML node (tag) / false on error

### 7.12.9 selectxmlcontent

#### 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!

#### Syntax

selectxmlcontent (\$xmldata, \$starttagname, \$startcondtag, \$condvalue)

- \$xmldata ... XML content container [string]
- \$starttagname ... tag name of requested XML node [string]
- \$startcondtag ... tag holding the conditional value inside the given starttagnamev
- \$condvalue ... conditional value [string]

result array with the content of the requested XML node (tag) / false on error

#### 7.12.10 selectxmlicontent

#### 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!

#### Syntax

selectxmlicontent (\$xmldata, \$starttagname, \$startcondtag, \$condvalue)

## Input parameters

- \$xmldata ... XML content container [string]
- \$starttagname ... tag name of requested XML node [string]
- \$startcondtag ... tag holding the conditional value inside the given starttagname [string]
- \$condvalue ... conditional value [string]

#### **Output**

result array with the content of the requested XML node (tag) / false on error

### 7.12.11 deletecontent

## 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.

#### **Syntax**

deletecontent (\$xmldata, \$starttagname, \$startcondtag="", \$condvalue="")

- \$xmldata ... XML content container [string]
- \$starttagname ... tag name of requested XML node [string]
- \$startcondtag ... tag holding the conditional value inside the given starttagname [string] (optional)

• \$condvalue ... conditional value [string] (optional)

## Output

XML content container / false on error

### 7.12.12 deleteicontent

## 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.

#### Syntax

deleteicontent (\$xmldata, \$starttagname, \$startcondtag="", \$condvalue="")

### Input parameters

- \$xmldata ... XML content container [string]
- \$starttagname ... tag name of requested XML node [string]
- \$startcondtag ... tag holding the conditional value inside the given starttagname [string] (optional)
- \$condvalue ... conditional value [string] (optional)

### Output

• XML content container / false on error

#### 7.12.13 setcontent

# 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.

#### Syntax

setcontent (\$xmldata, \$startparenttagname, \$starttagname, \$contentnew, \$startcondtag="", \$condvalue="")

### Input parameters

- \$xmldata ... XML content container [string]
- \$startparenttagname ... parent tag name [string]
- \$starttagname ... tag name of XML node for the new content [string]
- \$contentnew ... new XML node to be inserted
- \$startcondtag ... tag holding the conditional value inside the given starttagname [string]
- \$condvalue ... conditional value [string]

### Output

• XML content container / false on error

### 7.12.14 seticontent

#### 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.

## **Syntax**

seticontent (\$xmldata, \$startparenttagname, \$starttagname, \$contentnew, \$startcondtag, \$condvalue)

### Input parameters

- \$xmldata ... XML content container [string]
- \$startparenttagname ... parent tag name [string]
- \$starttagname ... tag name of XML node for the new content [string]
- \$contentnew ... new XML node to be inserted [string]
- \$startcondtag ... tag holding the conditional value inside the given starttagname [string]
- \$condvalue ... conditional value [string]

## Output

• XML content container / false on error

## 7.12.15 setcontent\_fast

### 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.

#### **Syntax**

setcontent\_fast (\$xmldata, \$startparenttagname, \$starttagname, \$contentnew, \$startcondtag="", \$condvalue="")

### Input parameters

- \$xmldata ... XML content container [string]
- \$startparenttagname ... parent tag name [string]
- \$starttagname ... tag name of XML node for the new content [string]
- \$contentnew ... new XML node to be inserted
- \$startcondtag ... tag holding the conditional value inside the given starttagname [string]
- \$condvalue ... conditional value [string]

### **Output**

• XML content container / false on error

## 7.12.16 updatecontent

# 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.

### **Syntax**

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

## Input parameters

- \$xmldata ... XML content container [string]
- \$xmlnode ... XML node to be replaced [string]
- \$xmInodenew ... new XML node [string]

#### Output

XML content container / false on error

## 7.12.17 insertcontent

# Description

<tagname> <- list start

.....

insertxmldata <- insertxmldata

</tagname> <- list end

......

Inserts \$insertxmldata string at the end of all child between the parent \$tagname .

#### **Syntax**

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

#### Input parameters

- \$xmldata ... XML content container [string]
- \$insertxmldata ... XML node to be inserted in starttagname [string]
- \$starttagname ... tag name of the parent XML node [string]

## Output

XML content container / false on error

## 7.12.18 inserticontent

#### **Description**

CASE-Insensitive version (XML parser are always case-sensitive!)

..... <tagname> <- list start

.....

 $insertxmldata <-\ insertxmldata$ 

</tagname> <- list end

Inserts \$insertxmldata string at the end of all child between the parent \$tagname.

#### **Syntax**

inserticontent (\$xmldata, \$insertxmldata, \$starttagname)

#### Input parameters

- \$xmldata ... XML content container [string]
- \$insertxmldata ... XML node to be inserted in starttagname [string]
- \$starttagname ... tag name of the parent XML node [string]

XML content container / false on error

#### 7.12.19 addcontent

# 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
```

#### **Syntax**

addcontent (\$xmldata, \$sub\_xmldata, \$startgrandtagname, \$startcondtag, \$condvalue, \$startparenttagname, \$starttagname, \$contentnew)

## Input parameters

- \$xmldata ... XML content container [string]
- \$sub\_xmldata ... xml node to be inserted [string]
- \$startgrandtagname ... grandparent tag name [string]
- \$startcondtag ... tag holding the conditional value inside the given starttagname [string]
- \$condvalue ... conditional value [string]
- \$startparenttagname ... parent tag name [string]
- \$starttagname ... tag name of XML node for the new content [string]
- \$contentnew ... new XML node to be inserted [string]

#### Output

XML content container / false on error

## 7.12.20 addicontent

#### 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
```

#### **Syntax**

addicontent (\$xmldata, \$sub\_xmldata, \$startgrandtagname, \$startcondtag, \$condvalue, \$startparenttagname, \$starttagname, \$contentnew)

#### Input parameters

- \$xmldata ... XML content container [string]
- \$sub\_xmldata ... xml node to be inserted [string]
- \$startgrandtagname ... grandparent tag name [string]
- \$startcondtag ... tag holding the conditional value inside the given starttagname [string]
- \$condvalue ... conditional value [string]
- \$startparenttagname ... parent tag name [string]
- \$starttagname ... tag name of XML node for the new content [string]
- \$contentnew ... new XML node to be inserted [string]

# Output

• XML content container / false on error

# 7.13 Report API Functions

## 7.13.1 createreport

#### **Description**

This function creates a new report

#### **Syntax**

createreport (\$report\_name)

## Input parameters

\$report\_name ... report name [string]

## global input parameters

- \$user
- \$eventsystem
- \$mgmt\_config
- \$hcms\_lang
- \$lang

## Output

result array

# 7.13.2 editreport

## Description

This function saves the configuration of a report.

## **Syntax**

editreport (\$report\_name, \$config)

## Input parameters

- \$report\_name ... report name [string]
- \$config ... report configuration [array]

## global input parameters

- \$user
- \$eventsystem
- \$mgmt\_config
- \$hcms\_lang
- \$lang

## Output

result array

## 7.13.3 loadreport

#### Description

This function loads the report configuration file and provides the data as array.

## **Syntax**

loadreport (\$report\_name)

#### Input parameters

• \$report\_name ... report name [string]

## global input parameters

- \$user
- \$eventsystem
- \$mgmt\_config
- \$hcms\_lang
- \$lang

## Output

result array / false

# 7.13.4 deletereport

## Description

This function deletes a report

## **Syntax**

deletereport (\$report\_name)

#### Input parameters

• \$report\_name ... report name [string]

## global input parameters

- \$user
- \$eventsystem
- \$mgmt\_config
- \$hcms\_lang
- \$lang

#### Output

result array

# 7.13.5 analyzeSQLselect

#### Description

This function analyzes an SQL Select statement and return its parts in an array

#### **Syntax**

analyzeSQLselect (\$sql)

## Input parameters

• \$sql ... SQL statement [string]

result array

## 7.13.6 showpiechart

#### **Description**

This function creates the Google Chart Code for a pie chart. The Google Chart API need to be loaded first!

#### **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 [string]
- \$x\_title ... chart x-axis title [string]
- \$x\_value ... chart x-axis array key name holding the values [string]
- \$y\_title ... chart y-axis title [string]
- \$y\_value ... chart y-axis array key name holding the values [string]
- \$data\_array ... assoz. data array [array]
- \$width ... chart width [string] (optional)
- \$height ... chart height [string] (optional)
- \$id ... ID of chart [string] (optional)
- \$suffix ... function name suffix [string] (optional)
- \$loadAPI ... load Google Chart API [boolean] (optional)

## global input parameters

- \$mgmt\_config
- \$hcms\_lang
- \$lang

## Output

· Google Chart code / false

#### 7.13.7 showcolumnchart

#### Description

This function creates the Google Chart Code for a column chart. The Google Chart API need to be loaded first!

## **Syntax**

```
show column chart ($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 [string]
- \$x\_title ... chart x-axis title [string]
- \$x\_value ... chart x-axis array key name holding the values [string]
- \$y1\_title ... chart y1-axis title [string]
- \$y1\_value ... chart y1-axis array key name holding the values [string]
- \$y2\_title ... chart y2-axis title [string]
- \$y2\_value ... chart y2-axis array key name holding the values [string]
- \$y3\_title ... chart y3-axis title
- \$y3\_value ... chart y3-axis array key name holding the values [string]
- \$data\_array ... assoz. data array [array]
- \$width ... chart width [string] (optional)
- \$height ... chart height [string] (optional)
- \$id ... ID of chart [string] (optional)
- \$suffix ... function name suffix [string] (optional)
- \$loadAPI

## global input parameters

- \$mgmt\_config
- \$hcms\_lang
- \$lang
- \$user

## Output

• Google Chart code / false

#### 7.13.8 showareachart

## Description

This function creates the Google Chart Code for an area chart. The Google Chart API need to be loaded first!

#### **Syntax**

```
showareachart ($title="", $x_title, $x_value, $y1_title, $y1_value, $y2_title="", $y2_value="", $y3_title="", $y3_value="", $data_array, $width="100%", $height="100%", $id="hcmsAreaChart", $suffix="", $loadAPI=true)
```

#### Input parameters

- \$title ... chart title [string]
- \$x\_title ... chart x-axis title [string]
- \$x\_value ... chart x-axis array key name holding the values [string]
- \$y1\_title ... chart y1-axis title [string]
- \$y1\_value ... chart y1-axis array key name holding the values [string]
- \$y2\_title ... chart y2-axis title [string]
- \$y2\_value ... chart y2-axis array key name holding the values [string]
- \$y3\_title ... chart y3-axis title
- \$y3\_value ... chart y3-axis array key name holding the values [string]
- \$data\_array ... assoz. data array [array]
- \$width ... chart width [string] (optional)
- \$height ... chart height [string] (optional)
- \$id ... ID of chart [string] (optional)
- \$suffix ... function name suffix [string] (optional)

\$loadAPI

## global input parameters

- \$mgmt\_config
- \$hcms\_lang
- \$lang
- \$user

## **Output**

Google Chart code / false

## 7.13.9 showtimelinechart

#### Description

This function creates the Google Chart Code for a timeline chart. The Google Chart API need to be loaded first!

#### **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 [string]
- \$y\_title ... chart y-axis title [string]
- \$y\_value ... chart y-axis array key name holding the values [string]
- \$x1\_title ... chart x1-axis title
- \$x1\_value ... chart x1-axis array key name holding the values [string]
- \$x2\_title ... chart x2-axis title [string]
- \$x2\_value ... chart x2-axis array key name holding the values [string]
- \$data\_array ... assoz. data array [array]
- \$width ... chart width [string] (optional)
- \$height ... chart height [string] (optional)
- \$id ... ID of chart [string] (optional)
- \$suffix ... function name suffix [string] (optional)
- \$loadAPI ... load Google Chart API [boolean] (optional)

## global input parameters

- \$mgmt\_config
- \$hcms\_lang
- \$lang
- \$user

#### Output

Google Chart code / false

## 7.13.10 showgeolocationchart

#### **Description**

This function creates the Google Maps Code for a geolocation chart. The Google Maps API need to be loaded first!

## **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 [string]
- \$marker\_value ... array key name holding the value for the marker title [string]
- \$lat\_value ... array key name holding the value for the latitude of the marker [string]
- \$Ing\_value ... array key name holding the value for the longitude of the marker [string]
- \$link\_value ... array key name holding the value for the marker link [string]
- \$data\_array ... assoz. data array [array]
- \$width ... chart width [string] (optional)
- \$height ... chart height [string] (optional)
- \$id ... ID of chart [string] (optional)
- \$suffix ... function name suffix [string] (optional)
- \$loadAPI ... load Google Maps API [boolean] (optional)

## global input parameters

- \$mgmt\_config
- \$hcms\_lang
- \$lang
- \$user

#### **Output**

Google Chart code / false

## 7.14 Project API Functions

## 7.14.1 createproject

#### Description

This function creates a new project.

#### **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) [integer]
- \$object\_id ... object ID or path to object [string] (optional)

- \$user ... user name of sub/project owner [string]
- \$projectname ... project name [string]
- \$description ... project description [string] (optional)

## global input parameters

- \$mgmt\_config
- \$hcms\_lang\_codepage
- \$hcms\_lang
- \$lang

## Output

result array

## 7.14.2 editproject

## Description

This function saves data of an exitsing project.

## **Syntax**

editproject (\$project\_id, \$subproject\_id=0, \$object\_id="", \$user="", \$projectname="", \$description="")

## Input parameters

- \$project\_id ... project ID [integer]
- \$subproject\_id ... ID of main project (only if project is a subproject) [integer]
- \$object\_id ... object ID or path to object [string] (optional)
- \$user ... user name of sub/project owner [string] (optional)
- \$projectname ... project name [string] (optional)
- \$description ... project description [string] (optional)

## global input parameters

- \$mgmt\_config
- \$hcms\_lang\_codepage
- \$hcms\_lang
- \$lang

#### **Output**

result array

## 7.14.3 deleteproject

## Description

This function removes projects.

## **Syntax**

deleteproject (\$project\_id)

## Input parameters

• \$project\_id ... project ID or array of project IDs to be deleted [string or array]

## global input parameters

- \$mgmt\_config
- \$hcms\_lang
- \$lang

## Output

result array

## 7.14.4 getprojectstructure

### **Description**

This function creates an associative array presenting the project structure (project -> subprojects -> tasks).

#### **Syntax**

getprojectstructure (\$project\_id, \$user="")

## Input parameters

- \$project\_id ... project ID [integer]
- \$user ... projects for specific user name [string] (optional)

## global input parameters

• \$mgmt\_config

## Output

associative result array / false

## 7.14.5 showprojectrecord

#### **Description**

This function creates a table row for the projects presentation.

## **Syntax**

showprojectrecord (\$project\_record, \$masterproject\_id)

#### Input parameters

- \$project\_record ... project record as array (project/subproject/task) [array]
- \$masterproject\_id ... ID of master project [integer]

## global input parameters

- \$mgmt\_config
- \$hcms\_lang
- \$lang
- \$siteaccess
- \$user\_array

## Output

· html presentation of the record / false

## 7.14.6 showganttchart

#### **Description**

This function creates the Google Chart Code for a Gantt chart. The Google Chart API need to be loaded first.

#### **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 [string]
- \$task\_title ... task title [string]
- \$task\_value ... task array key name holding the values [array]
- \$resource\_title ... resouce title [string] (optional)
- \$resource\_value ... resource array key name holding the values [string] (optional)
- \$start\_title ... task start\_date\_title [string]
- \$start\_value ... task start date array key name holding the values [string]
- \$end\_title ... task finish date title [string]
- \$end\_value ... task finish date array key name holding the values [string]
- \$status\_title ... work status title [string] (optional)
- \$status\_value ... work status array key name holding the values [string] (optional)
- \$data\_array ... assoz. data array [array]
- \$width ... chart width [string] (optional)
- \$height ... chart height [string] (optional)
- \$id ... ID of chart [string] (optional)
- \$suffix ... function name suffix [string] (optional)
- \$loadAPI ... load Google Chart API [boolean] (optional)

- \$mgmt\_config
- \$hcms\_lang
- \$lang

Google Chart code / false

## 7.15 Task API Functions

#### 7.15.1 createtask

## 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.

#### **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 [string] (optional)
- \$from\_user ... from\_user name [string]
- \$from\_email ... from\_email [email-address] (optional)
- \$to\_user ... to\_user name [string]
- \$to\_email ... to\_email [email-address] (optional)
- \$startdate ... start date [yyyy-mm-dd] (optional)
- \$finishdate ... finish date [yyyy-mm-dd] (optional)
- \$category ... category [link, user, workflow] (optional)
- \$object\_id ... object ID or object path [string]
- \$taskname ... task name [string]
- \$message ... message [string] (optional)
- \$sendmail ... sendmail [boolean]
- \$priority ... priority [high, medium, low] (optional)
- \$project\_id ... project/subproject ID if the task should be assigned to a project [integer] (optional)
- \$planned ... planned effort in taskunit [integer] (optional)

# global input parameters

- \$mgmt\_config
- \$hcms\_lang\_codepage
- \$hcms\_lang
- \$lang

## Output

result array

## 7.15.2 edittask

## 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.

#### **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 [integer]
- \$object\_id ... object ID pr path to object [string] (optional)
- \$to\_user ... to\_user name [string] (optional)
- \$startdate ... start date [yyyy-mm-dd] (optional)
- \$finishdate ... finish date [yyyy-mm-dd] (optional)
- \$taskname ... name of task [string] (optional)
- \$message ... task message/description [string] (optional)
- \$sendmail ... sendmail [true/false]
- \$priority ... priority [high, medium, low] (optional)
- \$status ... status in percent [0-100] (optional)
- \$planned ... planned effort in taskunit [integer] (optional)
- \$actual ... actual effort in taskunit [integer] (optional)
- \$project\_id ... project/subproject ID the task belongs to [integer] (optional)

#### global input parameters

- \$mgmt\_config
- \$hcms\_lang\_codepage
- \$hcms\_lang
- \$lang

# Output

result array

## 7.15.3 deletetask

## Description

Deletes user tasks.

## **Syntax**

deletetask (\$task\_id)

## Input parameters

• \$task\_id ... task ID or array of task IDs to be deleted [integer]

- \$mgmt\_config
- \$hcms\_lang
- \$lang

• true/false

#### 7.15.4 tasknotification

#### Description

Sends e-mail notifications to users if a task starts or ends on the given date.

## **Syntax**

tasknotification (\$date)

## Input parameters

• \$date ... date [yyyy-mm-dd]

## global input parameters

- \$mgmt\_config
- \$hcms\_lang\_codepage
- \$hcms\_lang
- \$lang

## Output

• true/false

# 7.16 Workflow API Functions

## 7.16.1 deleteworkflow

# Description

This function deletes a workflow.

#### Syntax

deleteworkflow (\$site, \$wf\_name, \$cat)

## Input parameters

- \$site ... publication name [string]
- \$wf\_name ... worfklow name [string]
- \$cat ... category [man, script]

- \$user
- \$eventsystem
- \$mgmt\_config
- \$hcms\_lang
- \$lang

result array

#### 7.16.2 buildworkflow

#### **Syntax**

buildworkflow (\$workflow\_data)

## Input parameters

• \$workflow\_data ... workflow [XML]

## Output

· workflow items as array

## 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 of executing user [string]
- \$message ... task message [string] (optional)
- \$sendmail ... sendmail [boolean] (optional)
- \$priority ... priority [high, medium, low] (optional)

- \$mgmt\_config
- \$hcms\_lang\_codepage
- \$hcms\_lang
- \$lang

workflow [XML-string] / false

## 7.16.4 acceptobject

#### **Description**

Accepts an object that is managed by a workflow.

### **Syntax**

acceptobject (\$site, \$location, \$object, \$item\_id, \$user, \$message="", \$sendmail=true, \$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 of executing user [string]
- \$message ... task message [string] (optional)
- \$sendmail ... sendmail [boolean] (optional)
- \$priority ... priority [high, medium, low] (optional)

## global input parameters

- \$mgmt\_config
- \$contentfile
- \$hcms\_lang\_codepage
- \$hcms\_lang
- \$lang

#### Output

result array

## 7.16.5 workflowreject

#### Description

Rejects an object that is managed by a workflow.

## **Syntax**

workflowreject (\$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 ... workflow item id [string]
- \$user ... user name of executing user [string]
- \$message ... task message [string] (optional)
- \$sendmail ... send mail [boolean] (optional)
- \$priority ... priority [high, medium, low] (optional)

## global input parameters

- \$mgmt\_config
- \$hcms\_lang
- \$lang

## Output

workflow [XML-string] / false

# 7.16.6 rejectobject

#### **Description**

Rejects an object that is managed by a workflow.

#### Syntax

rejectobject (\$site, \$location, \$object, \$item\_id, \$user, \$message="", \$sendmail=true, \$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 of executing user [string]
- \$message ... task message [string] (optional)
- \$sendmail ... send mail [boolean] (optional)
- \$priority ... priority [high, medium, low]

#### global input parameters

- \$mgmt\_config
- \$contentfile
- \$hcms\_lang\_codepage
- \$hcms\_lang
- \$lang

## Output

result array

# 7.17 Cloud Storage API Functions

## 7.17.1 loadcloudobject

#### **Description**

This functions load/streams an object/file from the cloud storage and saves it the local media repository on request

#### **Syntax**

loadcloudobject (\$site, \$location, \$file, \$user="", \$save=true, \$directoutput=false)

#### Input parameters

- \$site ... publication name [string]
- \$location ... media file location [string]
- \$file ... media file name [string]
- \$user ... user name [string] (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

# 7.17.2 savecloudobject

## 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']

## **Syntax**

savecloudobject (\$site, \$location, \$file, \$user="")

## Input parameters

- \$site ... publication name [string]
- \$location ... media file location [string]
- \$file ... media file name [string]
- \$user ... user name [string] (optional)

## global input parameters

• \$mgmt\_config

#### **Output**

• true / false on error

## 7.17.3 copycloudobject

#### **Description**

This functions renames an object/file in the cloud storage

#### **Syntax**

copycloudobject (\$site, \$location, \$file\_old, \$file\_new, \$user="")

## Input parameters

- \$site ... publication name [string]
- \$location ... media file location [string]
- \$file\_old ... source media file name [string]
- \$file\_new ... destination media file name [string]
- \$user ... user name [string] (optional)

## global input parameters

• \$mgmt\_config

## Output

• true / false on error

## 7.17.4 renamecloudobject

#### **Description**

This functions renames an object/file in the cloud storage

#### **Syntax**

renamecloudobject (\$site, \$location, \$file\_old, \$file\_new, \$user="")

#### Input parameters

- \$site ... publication name [string]
- \$location ... media file location [string]
- \$file\_old ... old media file name [string]
- \$file\_new ... new media file name [string]
- \$user ... user name [string] (optional)

## global input parameters

• \$mgmt\_config

## Output

• true / false on error

# 7.17.5 deletecloudobject

## Description

This function deletes an object/file and its data from the cloud storage

#### **Syntax**

deletecloudobject (\$site, \$location, \$file, \$user="")

## Input parameters

- \$site ... publication name [string]
- \$location ... media file location [string]
- \$file ... media file name [string]
- \$user ... user name [string] (optional)

## global input parameters

\$mgmt\_config

## Output

• true / false on error

## 7.17.6 synccloudobjects

## 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'].

#### Syntax

synccloudobjects (\$user="")

#### Input parameters

• \$user ... user name [string] (optional)

## global input parameters

• \$mgmt\_config

#### **Output**

• true / false on error

## 7.17.7 GCanalyzevideo

## **Description**

Analyzes a video and finds labels, based on Google Video Intelligence.

#### **Syntax**

GCanalyzevideo (\$site, \$path)

## Input parameters

- \$site ... publication name [string]
- \$path ... path or URI to a video file [string]

## global input parameters

\$mgmt\_config

## Output

• result array / false on error

# 7.17.8 GCanalyzeimage

## Description

Analyzes an image and finds labels, based on Google Vision.

#### **Syntax**

GCanalyzeimage (\$site, \$path)

#### Input parameters

- \$site ... publication name [string]
- \$path ... path or URI to a video file [string]

## global input parameters

• \$mgmt\_config

#### Output

• result array / false on error

## 7.17.9 GCspeech2text

#### Description

Analyzes a video or audio and extracts the text from the audio stream, based on Google Cloud Speech.

Google Speech supports the following language codes:

af-ZA, am-ET, hy-AM, az-AZ, id-ID, ms-MY, bn-BD, bn-IN, ca-ES, cs-CZ, da-DK, de-DE, en-AU, en-CA, en-GH, en-GB, en-IN, en-IE, en-KE, en-NZ, en-NG, en-PH, en-ZA, en-TZ, en-US, es-AR, es-BO, es-CL, es-CO, es-CR, es-EC, es-SV, es-ES, es-US, es-GT, es-HN, es-MX, es-NI, es-PA, es-PY, es-PE, es-PR, es-DO, es-UY, es-VE, eu-ES, fil-PH, fr-CA, fr-FR, gl-ES, ka-GE, gu-IN, hr-HR, zu-ZA, is-IS, it-IT, jv-ID, kn-IN, km-KH, lo-LA, lv-LV, lt-LT, hu-HU, ml-IN, mr-IN, nl-NL, ne-NP, nb-NO, pl-PL, pt-BR, pt-PT, ro-RO, si-LK, sk-SK, sl-SI, su-ID, sw-TZ, sw-KE, fi-FI, sv-SE, ta-IN, ta-SG, ta-LK, ta-MY, te-IN, vi-VN, tr-TR, ur-PK, ur-IN, el-GR, bg-BG, ru-RU, sr-RS, uk-UA, he-IL, ar-IL, ar-JO, ar-AE, ar-BH, ar-DZ, ar-SA, ar-IQ, ar-KW, ar-MA, ar-TN, ar-OM, ar-PS, ar-QA, ar-LB, ar-EG, fa-IR, hi-IN, th-TH, ko-KR, cmn-Hant-TW, yue-Hant-HK, ja-JP, cmn-Hans-HK, cmn-Hans-CN

#### **Syntax**

GCspeech2text (\$site, \$path, \$langcode="en-US")

## Input parameters

- \$site ... publication name [string]
- \$path ... path or URI to a video file [string]
- \$langcode ... language code [string] (optional)

## global input parameters

- \$mgmt\_config
- \$mgmt\_imagepreview
- \$mgmt\_mediapreview
- \$mgmt\_mediaoptions
- \$mgmt\_imageoptions
- \$mgmt\_maxsizepreview
- \$mgmt\_mediametadata
- \$mgmt\_compress
- \$hcms\_ext

#### **Output**

result array / false on error

# 7.18 Import/Export API Functions

## 7.18.1 exportobjects

## Description

Executes the export job.

#### **Syntax**

exportobjects ()

### Input parameters

#### global input parameters

• \$mgmt\_config

• true / false

# 7.18.2 importobjects

## Description

Executes the import job.

## **Syntax**

importobjects ()

## Input parameters

## global input parameters

• \$mgmt\_config

## Output

• true / false

# 7.18.3 importmediafiles

## Description

This function imports all directories and files from import directory to location.

#### **Syntax**

importmediafiles (\$importdir, \$location, \$createfolders=false, \$leavefiles=false, \$deletefiles=false, \$report=true, \$user="sys")

## Input parameters

- \$importdir ...
- \$location
- \$createfolders
- \$leavefiles
- \$deletefiles
- \$report
- \$user

## global input parameters

• \$mgmt\_config

## Output

true/false

## 7.18.4 importxmlfiles

## **Description**

Import of content and media files based on the XML import schema.

#### **Syntax**

importxmlfiles (\$importdir, \$createfolders=false, \$ignoremissinglinks=false, \$leavefiles=false, \$deletefiles=false, \$report=true, \$user="sys")

#### Input parameters

- \$importdir ... path of the import directory [string]
- \$createfolders ... create folder [boolean] (optional)
- \$ignoremissinglinks ... ignore or keep links to missing objects [boolean] (optional)
- \$leavefiles ... leave the files in the import directory and create symbolic link to the file [boolean] (optional)
- \$deletefiles ... delete files after successful import [boolean] (optional)
- \$report ... print report [boolean] (optional)
- \$user ... user name [string] (optional)

#### global input parameters

• \$mgmt\_config

## Output

true / false

#### 7.18.5 calculatedate

#### **Syntax**

calculatedate (\$days)

#### Input parameters

\$days ... number of days [integer]

## Output

date based on number of days in the past from today

## 7.18.6 exportxmlfiles

#### **Description**

Export of content and media files based on the XMLexport schema.

#### **Syntax**

exportxmlfiles (\$location, \$object="", \$exportdir, \$recursive=false, \$preserve=false, \$symlink=false, \$delete=false, \$createdays=0, \$editdays=0, \$accessdays=0, \$filesize=0, \$contentrules="", \$report=true, \$user="sys")

## Input parameters

- \$location ... location [string]
- \$object ... object name [string] (optional)
- \$exportdir ... path of the export directory or FTP server path [string]
- \$recursive ... export all items in subfolders recursively [boolean] (optional)
- \$preserve ... preserve container ID [boolean] (optional)
- \$symlink ... export only the media files and use a symbolic link to the file [boolean] (optional)
- \$delete ... delete files after successful export [boolean] (optional)
- \$createdays ... age in days after creation of the object [integer] (optional)
- \$editdays ... age in days after last modification of the object [integer] (optional)
- \$accessdays ... age in days after last access of the object [integer] (optional)
- \$filesize ... minimum file size of the asset in MB [integer] (optional)
- \$contentrules ... content rules/conditions for text content based on SQL Syntax [string]
- \$report ... print report [boolean] (optional)
- \$user ... user name [string] (optional)

## global input parameters

\$mgmt\_config

#### Output

true / false

# 8 Legal reference / flag

# 8.1 Questions and suggestions

For advanced questions and suggestions, please contact the support. We are available for every question regarding our reseller- and partner-program. You can apply for an access to our enhanced Online-Demo of the hyper Content & Digital Asset Management Servers via our support.

## hyperCMS Support:

support@hypercms.com
http://www.hypercms.com

# 8.2 Imprint

Responsible for the content:

hyperCMS Content Management Solutions GmbH Rembrandtstr. 35/6 A-1020 Vienna – Austria

office@hypercms.com http://www.hypercms.com

# 8.3 Legal information

The present product information is based on the version of the program, which was available at the time the document was composed.

The maker reserves the rights of modifications and corrections of the program. Errors and misapprehension accepted.

© 2019 by hyperCMS Content Management Solutions