

Versione 4.4.x

Di seguito vengono riportati dei templates di esempio relativi al lato client, che andranno opportunamente riconfigurati per la propria installazione. Occorre quindi tenere conto della struttura, ma revisionare tutti i singoli contenuti.

- [Client](#)
- [Admin](#)
- [Dettaglio Sezioni](#)

Client

[configuration.properties](#)

```

*****
#
#                               C L I E N T
#       C O N F I G U R A T I O N   P R O P E R T I E S
#                               T E M P L A T E
#
# in order to configure you application, has to be changed the
# sections between '<' and '>' characters
*****

#####
# B A S E P A T H
#####
basePath=file:///C:/Projects/<projectStaticFolder>/WEB/

#####
# T E M P P A T H
#####
#used by umplugin, scene, report
tempPath=C:/Project/<projectStaticFolder>/WEB/

#####
# F I L E S   C O M P R E S S I O N
#####
#useCompressed:  =false during development, =true when put in
production
useCompressed=false

#####
# M E T A D A T A   C A C H E   (W E B C L I E N T)
#####

```

```
#timeToIdleSeconds keep 1 in configuration phase. Increase as wanted in
production. tipycally 3600=1 hour
metadataacache.timeToIdleSeconds=1
#timeToLiveSecon keep 1 in configuration phase. Increase as wanted in
production. tipycally 3600=1 hour
metadataacache.timeToLiveSeconds=1

#####
# P R O J E C T   I N D E X
#####
applicationIndexTitle=Geoweb framework 4.4.0
#applicationIndexColor1=#00569F
applicationIndexColor1=#023B6E
applicationIndexColor2=white
applicationIndexColor3=#CECECE

#####
# G E N E R A L   S E T T I N G S
#####
#hide default detail report (STAMPA SCHEDA)
hideReportSingleView=true

#####
# M A P   G U I D E
#####
mapguide.mgUsername=Administrator
mapguide.mgPassword=admin
#mapguide library path. used to update mapguide layers (destination use
layer)
mapguide.library.path=Library://<projectStaticFolderName>

# webconfig.ini path
# Geoweb tries to load webconfig.ini firstly from static contents, and
secondarily from classPath.
# If all ways before fail, Geoweb uses this absolute path.
# This is generally used in Geoweb deployments that involve Linux +
WebLogic
pathTo.webconfig.ini=C:/Projects/<projectStaticFolder>/WEB/conf/webconf
ig.ini

#Next 4 parameters are used to dynamically overrides parameters set
inside webconfig.ini. They works in independent way each others. Remove
comment to override
#mapguide.webconfig.ini.AdministrativeConnectionProperties.Port=2810
#mapguide.webconfig.ini.ClientConnectionProperties.Port=2811
#mapguide.webconfig.ini.SiteConnectionProperties.Port=2812
```

```

#mapguide.webconfig.ini.SiteConnectionProperties.IpAddress=10.31.219.23
6

# resource package auto loading. Feature source
#Next 4, optional, parameters are used to auto configure feature source
database parameters
#during the procedure that applies resource package to all configured
MapGuide servers
#if not configured here the existing values inside .mpg file will not
to be overridden
#service often refers to the same machine in configured in jdbc.url,
but here MUST to be expressed like an IPV4 format address.
mapguide.resourcepackage.featuresource.service=<ip_address or
server_name>
mapguide.resourcepackage.featuresource.username=<schema_username>
mapguide.resourcepackage.featuresource.password=<schema_password>
mapguide.resourcepackage.featuresource.datastore=<DB_name>

#@Deprecated from 4.4.0 (but supported yet)
#mgUsername=
#mgPassword=
#geowebalias=
#pathTo.webconfig.ini=

#####
# A U T O M A T I C   D E P L O Y
#####
#geowebfolder contains resources to be automatically loaded during the
server startup
geowebfolder=C:\\projects\\<projectStaticFolder>\\geowebfolder

#####
# W O R K F L O W
#####
#This flag, default false, force to deploy all Process Definition
#inside Activiti at every server startup, even if already deployed.
#Usually all Activiti Process Definition, contained in one or more
#.bpmn files, under the folder [geowebfolder]/activiti/ will to be
#deployed inside Activiti, in a automatic way, during webapp
initialization
#If the Process Definition results already deployed, the deploy is
skipped
#This flag overrides this mechanism
workflow.activiti.forceDeploy.enabled=false

#####
# M A P

```

```
#####  
#Allowed range [0.001,1000] Ex: if map is in meter, but it had been  
loaded in mm, mapUnitFactor should be 1000  
mapUnitFactor=1  
  
#set the SRID used to handle spatial data from/to the database  
#this parameter SHOULD NOT generally to be set.  
#The SRID information is generally set inside MapGuide Maestro  
configuration  
#and it's enough. But sometimes spatial data in DB are populated in aim  
#to be consumed by other systems.  
#If omitted the default null SRID is used (change in base on DB vendor)  
#If set, this is used globally to hadle all spatial data  
#(but it can to be locally overridden using the gwAttribute xml  
parameter  
# named 'persistSridToDB')  
globalSRID=  
  
#####  
# U P L O A D   M A N A G E R  
#####  
  
#when loading a CAD drawing, specify, when true, if a SDF file is  
created for background layer, otherwise put all vector entities in  
drawing_background relations  
#useSDFForBackground=true  
useSDF=true  
  
# availablePackages=Bim\,PointCloud\,Plan\,BulkUp\,Views  
uploadManager.availablePackages=Bim\,PointCloud\,BulkUp\,Plan\,Views  
  
#projectType=building or model  
uploadManager.projectType=building  
  
# savingFileOn=db or alfresco (BulkUp - sperimentato su CDE)  
uploadManager.savingFileOn=alfresco  
#if alfresco -> set class, class attribute, code attribute,  
#column name of file name to search table record with the cmis widget  
uploadManager.fileClass=cde_deliverable_upload_manager  
uploadManager.fileAttribute=file_name  
uploadManager.fileCodeColumn=cod_deliverable_full  
uploadManager.fileName=cod_deliverable_full  
  
# Modello 3d (Bim,PointCloud)  
# saving IFC On=db or alfresco  
uploadManager.savingModelOn=alfresco  
uploadManager.modelClass=cde_deliverable_upload_manager
```

```
uploadManager.modelAttribute=file_name
uploadManager.modelCodeColumn=cod_deliverable_full
uploadManager.modelName=cod_deliverable_full

uploadManager.attachmentsAttribute=allegati_deliverable

#####
# GW  MAIL  CONF
#####
#from version 4.2.12 deprecates calendar.memoscheduler.* stuff

#email that will be shown as the sender by the email receiver. Used
when 'from' is not passed to gwMailService.sendMail(..)
gwMailService.defaultSenderEmail=no-reply@geowebitalia.it
gwMailService.user=demo.facilityservice@gmail.com
gwMailService.password=demoFS2016
gwMailService.host=smtp.gmail.com
gwMailService.port=465
#mail.smtp.starttls.enable - boolean - If true, enables the use of the
STARTTLS command (if supported by the server) to switch the connection
to a TLS-protected connection before issuing any login commands. Note
that an appropriate trust store must configured so that the client will
trust the server's certificate. Defaults to false.
gwMailService.mailSmtpStarttlsEnable=false
#mail.smtp.auth - boolean - If true, attempt to authenticate the user
using the AUTH command. Defaults to false.
gwMailService.mailSmtpAuth=true
#mail.smtp.socketFactory.port - int - Specifies the port to connect to
when using the specified socket factory. If not set, the default port
will be used.
gwMailService.mailSmtpSocketFactoryPort=465
#mail.smtp.socketFactory.class - String - If set, specifies the name of
a class that implements the javax.net.SocketFactory interface. This
class will be used to create SMTP sockets.
gwMailService.mailSmtpSocketFactoryClass=javax.net.ssl.SSLSocketFactory
#mail.smtp.socketFactory.fallback - boolean - If set to true, failure
to create a socket using the specified socket factory class will cause
the socket to be created using the java.net.Socket class. Defaults to
true.
gwMailService.mailSmtpSocketFactoryFallback=false
#mail.smtp.quitwait - boolean - If set to false, the QUIT command is
sent and the connection is immediately closed. If set to true (the
default), causes the transport to wait for the response to the QUIT
command.
gwMailService.mailSmtpQuitwait=false
#mail.debug - boolean
gwMailService.mailDebug=true
#mail.smtp.ssl.enable - boolean - If set to true, use SSL to connect
and use the SSL port by default. Defaults to false for the "smtp"
```

```
protocol and true for the "smtps" protocol.
gwMailService.mailSmtpSslEnable=
#mail.smtp.ssl.trust - String - If set, and a socket factory hasn't
been specified, enables use of a MailSSLConnectionFactory. If set to "*",
all hosts are trusted. If set to a whitespace separated list of hosts,
those hosts are trusted. Otherwise, trust depends on the certificate
the server presents.
gwMailService.mailSmtpSslTrust=*
#mail.smtp.ssl.protocols - String - Specifies the SSL protocols that
will be enabled for SSL connections. The property value is a whitespace
separated list of tokens acceptable to the
javax.net.ssl.SSLSocket.setEnabledProtocols method. (from gw version
4.4.17)
gwMailService.mailSmtpSslProtocols=

#####
# G W   S C H E D U L E R
#####

#toggle the gwSchedluer execution, optional, default true
gwscheduler.enabled=false
#toggle the support for cluster environment , optional, default false
#is subordinate to the addition of the field 'lock_end_time'
#to 'gw_sched_def' relation
gwscheduler.clusterMode.enabled=false

#####
# E V E N T   M E M O   S C H E D U L E R
#####

#toggle memo scheduler. When omitted default is true
eventmemo.scheduler.enabled=false

#http://www.quartz-scheduler.org/documentation/quartz-2.x/tutorials/cro
ntrigger.html
#
#      Field Name      Allowed Values      Allowed Special Characters
#      Seconds         0-59                , - * /
#      Minutes         0-59                , - * /
#      Hours           0-23                , - * /
#      Day-of-month    1-31                , - * ? / L W
#      Month           1-12 or JAN-DEC    , - * /
#      Day-of-Week     1-7 or SUN-SAT     , - * ? / L #
#
#1 0 3 * * ? stand for every day at 3:00:01 am
#calendar.memoscheduler.cronExpression=1 0 3 * * ?
eventmemo.scheduler.cronExpression=1 0 3 * * ?
```

```

#deleteEventMemoOfExpiredEventAfterEndAlert
eventmemo.scheduler.deleteEventMemoOfExpiredEventAfterEndAlert=false

#####
# MONITORING SCHEDULER
#####

#toggle monitoring scheduler. When omitted default is false
monitoring.scheduler.enabled=false

#http://www.quartz-scheduler.org/documentation/quartz-2.x/tutorials/crontrigger.html
#
#      Field Name      Allowed Values      Allowed Special Characters
#      Seconds         0-59                , - * /
#      Minutes         0-59                , - * /
#      Hours           0-23                , - * /
#      Day-of-month    1-31                , - * ? / L W
#      Month           1-12 or JAN-DEC     , - * /
#      Day-of-Week     1-7 or SUN-SAT      , - * ? / L #
#
#1 0 3 * * ? stand for every day at 3:00:01 am
#calendar.memoscheduler.cronExpression=1 0 3 * * ?
monitoring.scheduler.cronExpression=10 0 3 * * ?

#####
# PLANNED ACTIVITY SCHEDULER
#####

#toggle memo scheduler. When omitted default is false
plannedActivity.scheduler.enabled=false

#http://www.quartz-scheduler.org/documentation/quartz-2.x/tutorials/crontrigger.html
#
#      Field Name      Allowed Values      Allowed Special Characters
#      Seconds         0-59                , - * /
#      Minutes         0-59                , - * /
#      Hours           0-23                , - * /
#      Day-of-month    1-31                , - * ? / L W
#      Month           1-12 or JAN-DEC     , - * /
#      Day-of-Week     1-7 or SUN-SAT      , - * ? / L #
#
#1 0 3 * * ? stand for every day at 3:00:01 am
#calendar.memoscheduler.cronExpression=1 0 3 * * ?
plannedActivity.scheduler.cronExpression=20 0 3 * * ?

```

```
#####
# A E C  A S S I G N M E N T  S C H E D U L E R
#####

#toggle memo scheduler. When omitted default is false
calendar.aecAssignmentWS.enabled=false

#http://www.quartz-scheduler.org/documentation/quartz-2.x/tutorials/crontrigger.html
#
#      Field Name      Allowed Values      Allowed Special Characters
#      Seconds         0-59                , - * /
#      Minutes         0-59                , - * /
#      Hours           0-23                , - * /
#      Day-of-month    1-31                , - * ? / L W
#      Month           1-12 or JAN-DEC    , - * /
#      Day-of-Week     1-7 or SUN-SAT     , - * ? / L #
#
#1 0 3 * * ? stand for every day at 3:00:01 am
#calendar.memoscheduler.cronExpression=1 0 3 * * ?
calendar.aecAssignmentWS.cronExpression=1 0 22 * * ?

#####
# D A T A B A S E S
#####

#####
# database ORACLE
#####

#jdbc.driverClassName=oracle.jdbc.driver.OracleDriver
#jdbc.url=jdbc:oracle:thin:@<ip_address or
server_name>:<port>:<DB_name>
# es: jdbc.url=jdbc:oracle:thin:@ora11dev.gruppoesc.it:1521:ORA11DEV
#jdbc.username=<schemaData_username>
#jdbc.password=<schemaData_password>
#jdbc.maxActive=6
#jdbc.minIdle=2
#jdbc.maxIdle=6
#jdbc.validationQuery=select 1 from dual

#jdbcmetadata.driverClassName=oracle.jdbc.driver.OracleDriver
#jdbcmetadata.url=jdbc:oracle:thin:@<ip_address or
server_name>:<port>:<DB_name>
#es:
jdbcmetadata.url=jdbc:oracle:thin:@ora11dev.gruppoesc.it:1521:ORA11DEV
#jdbcmetadata.username=<schemaMetadata_username>
```



```

#jdbcmetadata.password=<schemaMetadata_password>
#jdbcmetadata.maxActive=6
#jdbcmetadata.minIdle=2
#jdbcmetadata.maxIdle=6
#jdbcmetadata.validationQuery=select 1 from dual

#####
# database POSTGRES
#####

jdbc.driverClassName=org.postgresql.Driver
jdbc.url=jdbc:postgresql://<ip_address or server_name>:<port>/<DB_name>
# es: jdbc.url=jdbc:postgresql://127.0.0.1:5432/TEST_GW44
jdbc.username=<schemaData_username>
jdbc.password=<schemaData_password>
jdbc.maxActive=6
jdbc.minIdle=2
jdbc.maxIdle=6
jdbc.validationQuery=select 1

jdbcmetadata.driverClassName=org.postgresql.Driver
jdbcmetadata.url=jdbc:postgresql://<ip_address or
server_name>:<port>/<DB_name>
# es: jdbcmetadata.url=jdbc:postgresql://127.0.0.1:5432/TEST_GW44
jdbcmetadata.username=<schemaMetadata_username>
jdbcmetadata.password=<schemaMetadata_password>
jdbcmetadata.maxActive=6
jdbcmetadata.minIdle=2
jdbcmetadata.maxIdle=6
jdbcmetadata.validationQuery=select 1

#####
# database SQLSERVER
#####

#jdbc.driverClassName=com.microsoft.sqlserver.jdbc.SQLServerDriver
#jdbc.url=jdbc:sqlserver://<ip_address or
server_name>:<port>;databaseName=<DB_name>
# es: jdbc.url=jdbc:sqlserver://192.168.0.99:1433;databaseName=geoweb
#jdbc.username=<schemaData_username>
#jdbc.password=<schemaData_password>
#jdbc.maxActive=6
#jdbc.minIdle=2
#jdbc.maxIdle=6
#jdbc.validationQuery=select 1
#
#jdbcmetadata.driverClassName=com.microsoft.sqlserver.jdbc.SQLServerDri
ver
#jdbcmetadata.url=jdbc:sqlserver://<ip_address or

```

```
server_name>:<port>;databaseName=<DB_name>
# es:
jdbcmetadata.url=jdbc:sqlserver://192.168.0.99:1433;databaseName=geoweb
jdbcmetadata.username=<schemaData_username>
jdbcmetadata.password=<schemaData_password>
jdbcmetadata.maxActive=6
jdbcmetadata.minIdle=2
jdbcmetadata.maxIdle=6
jdbcmetadata.validationQuery=select 1

#####
# M O N G O D B
#####
# Convention: for each mongo.db_* use as value use this pattern: (since
4.4.0).
#     [DATABASE_NAME][SUFFIX]
#     available suffixes: [_BIM_MODELS, _CLASSIFICATION, _MNEMONIC_CODE]
#
# Example (if DATABASE_NAME=DB_NAME):
#     mongo.db_classification=DB_NAME_CLASSIFICATION

mongo.db_bim_models=<MongoDB_name>_BIM_MODELS
mongo.db_classification=<MongoDB_name>_CLASSIFICATION
mongo.db_mnemonic_code=<MongoDB_name>_MNEMONIC_CODE

#mongo credentials. optional parameters (since 4.4.0)
#
# mongo.authentication optional, default null, allowed values: [SCRAM-
SHA-1,MONGODB-X509,MONGODB-CR,GSSAPI,PLAIN]
# mongo.password optional, default null. Depending on
mongo.authentication value, password can be required or forbidden
#     when mongo.authentication value is SCRAM-SHA-1 or MONGODB-CR
or PLAIN => password must be always configured
#     when mongo.authentication value is GSSAPI or MONGODB-X509 =>
password must be always omitted
# mongo.username is always required if credentials are enabled
# mongo.source is always required if credentials are enabled
# Credentials are considered enabled if at least one of these parameter
is configured. Geoweb will try to create the credentials for the
# MongoClientFactoryBean bean

mongo.source=<MongoDB_name>
mongo.username=<schemaMongo_username>
mongo.password=<schemaMongo_username>
mongo.authentication=SCRAM-SHA-1

#mongo.uri
# optional, default null (since 4.4.0). It is alternative to both
```

```

mongo.host and mongo.port and it has the evaluation priority upon them
# Inside mongo.uri can be specified multiple host, eventually each one
with it's port (default is 27017)
# Inside mongo.uri can be specified some credentials. These credentials
are used if the flag mongo.forceToUseURICredentials=true
# In that case even the credentials deriving from
mongo.username/mongo.password are overridden.
# Inside mongo.uri can be specified some options: they are applied
# Inside mongo.uri can be specified database
#
# mongo.uri pattern:
#
mongodb://[username:password@]host1[:port1][,host2[:port2],...[,hostN[:
portN]]][/[database][?options]]
#
# mongo.uri examples:
#   mongodb://localhost
#   mongodb://localhost1:27017
#   mongodb://localhost1:27018
#
mongodb://username:password@localhost:27017,localhost1,localhost2:27018

```

mongo.uri=mongodb://<ip_address or server_name>:<port>

#mongo.forceToUseURICredentials, evaluated only if mongo.uri is configured, default false (since 4.4.0)

mongo.forceToUseURICredentials=false

#mongo.host and mongo.port are alternative to mongo.uri

#mongo.host=<ip_address or server_name>

#mongo.port=<port>

#####

C M I S

#####

#ALFRESCO

cmisUser=admin

cmisPassword=<cmisAdminPassword>

cmisUrl=http://127.0.0.1:8081/alfresco/service/cmis

#la cartella su Alfresco va creata manualmente accedendo ad Alfresco e posizionandola sotto lo spazio 'Sites/Siti'

cmisBasePath=/Siti/Geoweb

cmisType=alfresco

#SHAREPOINT

#cmisUser=geowebitalia\\sharepoint

#cmisPassword=Shar3point\$

#cmisUrl=http://winsharepoint/_vti_bin/cmis/rest?getRepositories

```
#cmisBasePath=/
#cmisType=sharepoint

#####
# S E C U R I T Y
#####

# length of the time period (in days) for which password is valid -
Periodo di validità delle password
gwSecurity.passwordLifeTimeDays=180
# threshold to prohibit reuse of password up to a previous date -
Numero minimo di password differenti prima del riutilizzo
gwSecurity.passwordHistoricalCheckingCount=10
# threshold (in days) to prohibit the reuse of the password used from a
date onwards
gwSecurity.passwordHistoricalCheckingPeriodDays=1800
# (Account lockout threshold) number of invalid logon attempts that are
allowed before an account is locked out. - Numero massimo di tentativi
di accesso con password errata
gwSecurity.maxAttempts=10
# Account locked time
gwSecurity.accountLockedTimeMinutes=30
# rest token lifetime for reset password ('forget me')
gwSecurity.tokenLifeTimeHours=24

# email forgot password - Subjet - password dimenticata
gwSecurity.resetPasswordEmailSubjet=Ambiente di TEST GW4.4 -
Impostazione/Ripristino Dati di Accesso

# email forgot password - text message - params: {0: userEmail, 1:
username, 2: reset_token, 3: resetUrl}
# gwSecurity.resetPasswordEmailMessage=Gentile {1}\, <br>per inserire
una nuova password\, clicca sul link sottostante:<br> {3} <br><br><br>
Cordiali saluti\, <br> Geowebitalia
gwSecurity.resetPasswordEmailMessage=Gentile {1}\, <br>E stata inviata
una richiesta di impostazione o ripristino della password per il tuo
account sul sistema GW4.4 on CDE.<br><br>Puoi ora accedere cliccando
sul seguente collegamento o copiandolo e incollandolo nel browser:
<br><a href="{3}">{3}</a> <br><br>Questo collegamento puo essere
utilizzato per accedere una sola volta\, verrai indirizzato a una
pagina dove potrai impostare la tua password. <br>Scade dopo un giorno
e non succede nulla se non viene utilizzato. <br><br><br>Cordiali
saluti

#####
# C L I E N T   P A S S W O R D   R E G E X
#####
```

```

#gwClientPasswordRegex: regex applied to dijit/form/ValidationTextBox
widget
#gwClientPasswordRegex=^(?=.*?[A-Z])(?=.*?[a-
z])(?=.*?[0-9])(?=.*?[#?!@$%^&*-_]).{8\|,}$

#Minimum eight characters, at least one uppercase letter, one lowercase
letter, one number and one special character:
gwClientPasswordRegex=^(?=.*[a-z])(?=.*[A-
Z])(?=.*\\d)(?=.*[$@!%*?&])[A-Za-z\\d@$!%*?&]{8\|,}

#gwClientPasswordRegexPromptMessage: Tooltip text that appears when the
text box is empty and on focus
gwClientPasswordRegexPromptMessage=La password deve contenere almeno 8
caratteri di cui almeno una lettera maiuscola\, almeno una lettera
minuscola\, almeno un numero\, almeno un carattere speciale fra
$!%*?&
##?!@$%^&_-

#####
# A U T H E N T I C A T I O N
#####

#PRE AUTHENTICATION CREDENTIALS (optional)
#Next two parameters, both optional, work in conjunction and maybe used
to create a simplified authentication mechanism (if oauth2 is not
possible, or not jet implemented, for example)
#Every call to server will be considered allowed if contains in its
headers section a parameter called SM_USER properly populated.
#This parameter should be computed making the sha512 of the string
resulting from the concatenation of preAuthUser and preAuthKey
parameter, with a | character in the middle.
#Here the pseudo code:
#
# var SM_USER = SHA512(preAuthUser+"|"+preAuthKey)
#
# Ex:
#
#     SM_USER:
2ca0e5a3633f7c8306505d3c7edcdaac29c93ae689e0b1182c3da4bfdc763758745e768
4926042b840d6beb193ffd4e11fa9d1d73d0bec43f42348cca4f2aedd
#
#preAuthUser=ws_test
#preAuthKey=ws_test_1

#PRE AUTHENTICATED LOGOUT (optional)
#By default if an user click logout button, Geoweb redirect him to
login page. The behavior can be overridden setting this variable: user
will be redirected at set url
#preAuthenticated_logout_url=https://loginatla.gse.it/AGLogout

```

```
#####  
# LOGOS CUSTOMIZATION  
#####  
#CUSTOM INDEX BACKGROUND PATH: WEB\; example:  
#images.IndexBackground=MyCostumImage.jpg  
#CUSTOM LOGIN LOGO PATH: WEB\IMAGES; example:  
#images.login.form.logo=MyCustomLogo.png  
#CUSTOM INDEX BACKGROUND PATH: WEB\; example:  
#images.bannerLeft=MyCustomBannerLeft.jpg  
#images.bannerCenter=MyCustomBannerCenter.png  
#images.bannerRight=MyCustomBannerRight.jpeg
```

Admin

configuration.properties

```
#####  
# ADMIN  
# CONFIGURATION PROPERTIES  
# TEMPLATE  
#  
# in order to configure you application, has to be changed the  
# sections between '<' and '>' characters  
#####  
  
#####  
# BASE PATH  
#####  
basePath=file:///C:/Projects/<projectStaticFolder>/WEB/  
  
#####  
# TEMP PATH  
#####  
#used by umplugin, scene, report  
tempPath=C:/Project/<projectStaticFolder>/WEB/  
  
#####  
# FILES COMPRESSION  
#####  
#useCompressed: =false during development, =true when put in  
production  
useCompressed=true
```

```
#####
# M E T A D A T A   C A C H E (WEBADMIN)
#####

#comma (,) separated client server address. Only one is the common case
(no server cluster)
urlList=http://localhost/<projectStaticFolder>/

#####
# D A T A B A S E S
#####

#####
# database ORACLE
#####

#jdbc.driverClassName=oracle.jdbc.driver.OracleDriver
#jdbc.url=jdbc:oracle:thin:@<ip_address or
server_name>:<port>:<DB_name>
# es: jdbc.url=jdbc:oracle:thin:@ora11dev.gruppoesc.it:1521:ORA11DEV
#jdbc.username=<schemaData_username>
#jdbc.password=<schemaData_password>
#jdbc.maxActive=6
#jdbc.minIdle=2
#jdbc.maxIdle=6
#jdbc.validationQuery=select 1 from dual

#jdbcmetadata.driverClassName=oracle.jdbc.driver.OracleDriver
#jdbcmetadata.url=jdbc:oracle:thin:@<ip_address or
server_name>:<port>:<DB_name>
#es:
jdbcmetadata.url=jdbc:oracle:thin:@ora11dev.gruppoesc.it:1521:ORA11DEV
#jdbcmetadata.username=<schemaMetadata_username>
#jdbcmetadata.password=<schemaMetadata_password>
#jdbcmetadata.maxActive=6
#jdbcmetadata.minIdle=2
#jdbcmetadata.maxIdle=6
#jdbcmetadata.validationQuery=select 1 from dual

#####
# database POSTGRES
#####

jdbc.driverClassName=org.postgresql.Driver
jdbc.url=jdbc:postgresql://<ip_address or server_name>:<port>/<DB_name>
# es: jdbc.url=jdbc:postgresql://127.0.0.1:5432/TEST_GW44
jdbc.username=<schemaData_username>
jdbc.password=<schemaData_password>
```

```
jdbc.maxActive=6
jdbc.minIdle=2
jdbc.maxIdle=6
jdbc.validationQuery=select 1

jdbcmetadata.driverClassName=org.postgresql.Driver
jdbcmetadata.url=jdbc:postgresql://<ip_address or
server_name>:<port>/<DB_name>
# es: jdbcmetadata.url=jdbc:postgresql://127.0.0.1:5432/TEST_GW44
jdbcmetadata.username=<schemaMetadata_username>
jdbcmetadata.password=<schemaMetadata_password>
jdbcmetadata.maxActive=6
jdbcmetadata.minIdle=2
jdbcmetadata.maxIdle=6
jdbcmetadata.validationQuery=select 1

#####
# database SQLSERVER
#####

#jdbc.driverClassName=com.microsoft.sqlserver.jdbc.SQLServerDriver
#jdbc.url=jdbc:sqlserver://<ip_address or
server_name>:<port>;databaseName=<DB_name>
# es: jdbc.url=jdbc:sqlserver://192.168.0.99:1433;databaseName=geoweb
#jdbc.username=<schemaData_username>
#jdbc.password=<schemaData_password>
#jdbc.maxActive=6
#jdbc.minIdle=2
#jdbc.maxIdle=6
#jdbc.validationQuery=select 1
#
#jdbcmetadata.driverClassName=com.microsoft.sqlserver.jdbc.SQLServerDri
ver
#jdbcmetadata.url=jdbc:sqlserver://<ip_address or
server_name>:<port>;databaseName=<DB_name>
# es:
jdbcmetadata.url=jdbc:sqlserver://192.168.0.99:1433;databaseName=geoweb
#jdbcmetadata.username=<schemaData_username>
#jdbcmetadata.password=<schemaData_password>
#jdbcmetadata.maxActive=6
#jdbcmetadata.minIdle=2
#jdbcmetadata.maxIdle=6
#jdbcmetadata.validationQuery=select 1

#####
# S E C U R I T Y
#####
```



```

# length of the time period (in days) for which password is valid -
Periodo di validità delle password
gwSecurity.passwordLifeTimeDays=180
# threshold to prohibit reuse of password up to a previous date -
Numero minimo di password differenti prima del riutilizzo
gwSecurity.passwordHistoricalCheckingCount=10
# threshold (in days) to prohibit the reuse of the password used from a
date onwards
gwSecurity.passwordHistoricalCheckingPeriodDays=1800
# (Account lockout threshold) number of invalid login attempts that are
allowed before an account is locked out. - Numero massimo di tentativi
di accesso con password errata
gwSecurity.maxAttempts=10
# Account locked time
gwSecurity.accountLockedTimeMinutes=30
# rest token lifetime for reset password ('forget me')
gwSecurity.tokenLifeTimeHours=24

# email forgot password - Subjet - password dimenticata
gwSecurity.resetPasswordEmailSubjet=Richiesta cambio password

# email forgot password - text message - params: {0: userEmail, 1:
username, 2: reset_token, 3: resetUrl}
# gwSecurity.resetPasswordEmailMessage=Gentile {1}\,\,<br>per inserire
una nuova password\,\, clicca sul link sottostante:<br> {3} <br><br><br>
Cordiali saluti\,\, <br> Geowebitalia
gwSecurity.resetPasswordEmailMessage=Gentile {1}\,\,<br>E stata inviata
una richiesta di impostazione o ripristino della password per il tuo
account sul sistema GW4.<br><br>Puoi ora accedere cliccando sul
seguito collegamento o copiandolo e incollandolo nel browser: <br><a
href="{3}">{3}</a> <br><br>Questo collegamento puo essere utilizzato
per accedere una sola volta\,\, verrai indirizzato a una pagina dove
potrai impostare la tua password. <br>Scade dopo un giorno e non
succede nulla se non viene utilizzato. <br><br><br>Cordiali saluti

#####
# C L I E N T   P A S S W O R D   R E G E X
#####
#gwClientPasswordRegex: regex applied to dijit/form/ValidationTextBox
widget (optional)
#gwClientPasswordRegex=^(?=.*?[A-Z])(?=.*?[a-
z])(?=.*?[0-9])(?=.*?[#?!@$%^&*-_]).{8\,\,}$

#Minimum eight characters, at least one uppercase letter, one lowercase
letter, one number and one special character:
gwClientPasswordRegex=^(?=.*?[a-z])(?=.*?[A-
Z])(?=.*?\d)(?=.*?[$@!%*?&])[A-Za-z-\d$@!%*?&]{8\,\,}

#gwClientPasswordRegexPromptMessage: Tooltip text that appears when the

```

```
text box is empty and on focus (optional)
#gwClientPasswordRegexPromptMessage=La password deve contenere almeno 8
caratteri di cui almeno una lettera maiuscola\\, almeno una lettera
minuscola\\, almeno un numero\\, almeno un carattere speciale fra
#?!@$%^&-_
gwClientPasswordRegexPromptMessage=La password deve contenere almeno 8
caratteri di cui almeno una lettera maiuscola\\, almeno una lettera
minuscola\\, almeno un numero\\, almeno un carattere speciale fra
$!*?&#
#?!@$%^&-_
```

Dettaglio Sezioni

MongoDB

Dentro dispatcher-servlet.xml

Dalla versione 4.4.x la non è più necessario intervenire nel file dispatcher-servlet.xml per la configurazione dBMongo DB. Deve solo rimanere dichiarato questo bean:

```
<bean id="mongoClientFactoryBean"
class="com.geowebframework.dataservice.mongo.GwMongoClientFactoryBean">
</bean>
```

Bean con commento:

```
<!-- MONGODB CONFIGURATION -->
<!--
    GwMongoClientFactoryBean is a custom
    org.springframework.data.mongodb.core.MongoClientFactoryBean
    that handles more than one single mongo instance (defined by
    host:port). It handles a replica set.
    Replica set in MongoDB is a group of mongodb processes that maintain
    the same data set.

    evaluated keys inside configuration.properties file
    mongo.uri
    mongo.forceToUseURICredentials
    mongo.host
    mongo.port

    It, respect the vanilla MongoClientFactoryBean, handles the variable
    'uri'.
    Parameter 'uri' has evaluation priority upon 'host'+ 'port'
    parameters.
```

When using 'uri' property, credentials can be optionally provided as an uri parameter (in queryString) named 'credentials'.

If mongo.forceToUseURICredentials flag is provided and set to true the credentials, if available, are extrapolated from uri and applied to mongoClientFactoryBean

If at least one of the properties below in configuration.properties file are configured

mongo credentials are automatically created and applied to mongoClientFactoryBean

(not all are required, depending on mongo.authentication value)

mongo.username

mongo.password

mongo.source

mongo.authentication

-->

```
<bean id="mongoClientFactoryBean"  
class="com.geowebframework.dataservice.mongo.GwMongoClientFactoryBean">  
</bean>
```

Tutte le configurazioni sono state rese di fatto manovrabili dal file configuration.properties.

mongo.uri

Il parametro mongo.uri, disponibile dalla 4.4.x, è un'alternativa all'utilizzo di entrambi i parametri mongo.host e mongo.port, ed ha priorità di valutazione rispetto ad essi. Dentro mongo.uri si possono specificare più host, ognuno eventualmente corredato dalla propria porta (di default è 27017). Dentro mongo.uri possono essere specificate delle credenziali. Queste credenziali sono utilizzate quando il flag mongo.forceToUseURICredentials=true. Dentro mongo.uri possono essere specificate ulteriori opzioni. Esse sono in genere gestite dal bean che effettua la connessione. Dentro mongo.uri può essere specificato il database.

mongo.uri pattern:

```
mongodb://[username:password@]host1[:port1][,host2[:port2],...[,hostN[:portN  
]][/database][?options]
```

mongo.uri esempi:

```
mongodb://localhost  
mongodb://localhost:27017  
mongodb://localhost:27018  
mongodb://username:password@localhost:27017,localhost1,localhost2:27018
```

Mongo Credentials

L'istanza del DB di Mongo può prevedere l'abilitazione di credenziali (assolutamente consigliato in produzione). Questo è il set di parametri, opzionali, che va eventualmente configurato:

- **mongo.authentication** optional, default null, allowed values: [SCRAM-SHA-1,MONGODB-X509,MONGODB-CR,GSSAPI,PLAIN]
- **mongo.password** optional, default null. Depending on mongo.authentication value, password can be required or forbidden:
 - when mongo.authentication value is SCRAM-SHA-1 or MONGODB-CR or PLAIN ⇒ password must be always configured
 - when mongo.authentication value is GSSAPI or MONGODB-X509 ⇒ password must be always omitted
- **mongo.username** is always required if credentials are enabled
- **mongo.source** is always required if credentials are enabled

L'utilizzo delle credenziali si considera attivato se almeno uno di questi parametri è configurato. Geoweb proverà ad utilizzarli per creare le credenziali per il bean mongoClientFactoryBean. Da notare che queste possono essere sovrascritte dalle credenziali eventualmente present dentro mongo.uri, se il flag mongo.forceToUseURICredentials è abilitato.

Mongo Database

Geoweb si aspetta esistere tre database, che vengono usati per persistere i dati riguardanti 3 macrocomponenti del framework:

- **Modelli BIM**
- **Classificazione**
- **Codici Mnemonici**

Per convenzione ogni proprietà mongo.db_* del configuration.properties va configurata seguendo questo pattern:

```
[DATABASE_NAME] [SUFFIX]
```

Dove i suffissi disponibili sono:

- '_BIM_MODELS'
- '_CLASSIFICATION'
- '_MNEMONIC_CODE'

Per esempio, se [DATABASE_NAME] vale 'DB_NAME':

```
mongo.db_classification=DB_NAME_CLASSIFICATION
```

Gestori Documentali (CMIS)

TODO

MapGuide

MgInitializer

Service that initialize MapGuide WebTier using the provided webconfig.ini file.

Geoweb loading strategy

- firstly is used the `WEBCONFIG_INI_PATH` absolute path configured inside the `configuration.properties` file (This is generally used in deployments where a dynamic behavior is needed, or that involves Linux + WebLogic, or Docker-like contexts)
- secondarily from static contents
- as last chance from `classPath`

Re-mapping

The `remapConfiguration.properties` mechanism that load properties from System Environment/Context variables is supported as well, and is often used when dynamic setting is needed.

So the path in `WEBCONFIG_INI_PATH` can be dynamically re-mapped to a System Environment/Context variable (configuring a mapping inside `remapConfiguration.properties`).

There is the possibility to dynamically override also the `IpAddress` and `Port(s)` properties set inside `webconfig.ini` file, using the appropriate configurations existing inside `configuration.properties` file (see all the `mapguide.webconfig.ini.*` properties below).

If at least one of the `IpAddress` and `Port(s)` properties is overridden. A new `webconfig.ini` file (named `webconfig_mod.ini`) will be created (at the same path) with the new configurations and given to be processed by the `MapGuideJavaApi.MgInitializeWebTier(path)` java API. The `IpAddress` is handled by MapGuide only in the IPV4 form, but it can come from `configuration.properties/SystemEnvVariables` as a IPV6 or a host name. In case of host name, it will be resolved using a DNS service, converted in a IPV4 format address. In case of IPV6 address, it will be converted in the equivalent IPV4, only when possible. In all cases the new IP address will be set into the new `webconfig_mod.ini` file in IPV4 format. `IpAddress` can be composed by more than one IP address, comma separated (','), in all the supported formats above. If more than one IP address is configured, all the invalid ones will be not used, logging the fact. MapGuide will be however initialized with the remaining valid IP addresses.

Automatic resource package load

Before version 4.4.0 it was possible to export/import inside MapguideServer the `.mpg` resources files

only using MapGuide Maestro. Now there is an automatic procedure that look for all the packages with extension .mpg existing under this path [geowebfolder]/mapguide/ The [geowebfolder] is configurable under configuration.properties. If the file is found, GeoWeb try to load to all the available .mpg files to all the MapGuide servers. MapGuide servers list is retrieved from the IpAddress property inside section SiteConnectionProperties inside webconfig.ini file. If more than one is configured, Geoweb expects them as a comma separated ('.') list. Each packages can contains one or more folder with a resources structure. There is a mechanism that ensures that a single .mpg file, if not modified subsequently, it will not to be reloaded on MapGuide on the next server startup (the md5 of file is saved to check if file has changed at the next time). To perform this check there an improper use of a MapGuide metadata: Symbol. We choose to use a MapGuide metadata, like Symbols, because they are easily accessible through Java API from all the eventual existing Geoweb instances, that share the use of the one(multiple) MapGuide server(s). All Symbols used to persist loading informations are stored under a support folder called '_gw_packages_info', under the main MapGuide Library folder. A Symbol is created for each .mpg file. As 'Name' is used the .mpg file, without extension. 'Description' field is populated concatenating [md5hex]+'_@@@_'+[true/false]. The final boolean is a flag used to persist the 'isLoading' status during the loading resources process. If a Geoweb instance finds this flag set to true, is notified that another Geoweb instance is taking care of loading that resource package, and it will not care about doing it.

Automatic resource package processing

For every loading resource .mpg packages deeper elaborations maybe performed. All files *.FeatureSource_CONTENT.xml may to be modified if as <Provider> declare one the the handled types:

- King.Oracle
- OSGeo.PostgreSQL
- OSGeo.SQLServerSpatial

On all matching files inside .mpg package is performed a substitution regarding the content of these tags:

- <Service>
- <Username>
- <Password>
- <DataStore>

The new tag content is retrieved from some specific properties inside configuration.properties file with this radix:

```
mapguide.resourcepackage.featuresource.*
```

All these properties support the remapConfiguration.properties mechanism that load properties from System Environment/Context variables.

Note: is important to be sure that the .mpg file will not to be used by another process at the Geoweb server startup (7zip for example). In that case Geoweb will not to be able to process the file, and an exception will be thrown. The exception will not to be blocking. So in case of multiple .mpg files each loading process will continue independently.

Esempi Log

Ecco alcuni esempi del log riepilogativo che viene prodotto in fase di startup del server Geoweb.

Un solo IpAddress configurato; un file .mpg trovato; elaborazione .mpg terminata con successo; risorsa caricata con successo;

```
-----  
-----  
15-10-2019 10:18:18,085 [localhost-startStop-1] GEOWEB INFO  
  
MgInizializer - init()  
Now Geoweb wil try to load the webconfig.ini. This is the loading priority.  
Geoweb firstly look for the property mapguide.pathTo.webconfig.ini inside  
configuration.properties. Secundarily, if missing, Geoweb looks for the file  
webconfig.ini under static contents folder.As last chance Geoweb looks for  
the file webconfig.ini under the classPath.  
Property mapguide.pathTo.webconfig.ini had been found inside  
configuration.properties file with valorized  
S:/deploy/GW_MODULES_MG31/conf/webconfig.ini  
Configured path refers to an existing file  
Initializing MapGuide with the file:  
S:/deploy/GW_MODULES_MG31/conf/webconfig.ini  
applyResourcePackageInAllMapGuideServer()  
Checking if there is a resource package to apply to MapGuide Server..  
Looking for folder S:\deploy\GW_MODULES_MG31\WEB\geowebfolder\mapguide..  
Folder S:\deploy\GW_MODULES_MG31\WEB\geowebfolder\mapguide exists  
building all the resource package in folder:  
S:\deploy\GW_MODULES_MG31\WEB\geowebfolder\mapguide  
building resource package named: cde_test_2.mpg  
siteAddressList: [192.168.0.225]  
applyResourcePackage() - resourcePackage:  
S:\deploy\GW_MODULES_MG31\WEB\geowebfolder\mapguide\cde_test_2.mpg  
computeMd5Hex() - file:  
S:\deploy\GW_MODULES_MG31\WEB\geowebfolder\mapguide\cde_test_2.mpg  
md5 computed in nanoseconds: 12289400  
computed md5Hex: 19f5ddb761d9bb4dc3668e5a1df234df  
  * nAddressLeft: 1  
  * attemp number: 1  
  * currentSiteAddress: 192.168.0.225  
  * Resource package correctly applied to MapGuide site address:  
192.168.0.225  
done  
-----  
-----
```

Due IpAddress configurati (uno esistente ed uno finto); un file .mpg trovato; elaborazione .mpg terminata con successo; risorsa caricata con successo solo su un MapGuide address;

```
-----  
15-10-2019 10:18:18,085 [localhost-startStop-1] GEOWEB INFO  
  
MgInizializer - init()  
Now Geoweb wil try to load the webconfig.ini. This is the loading priority.  
Geoweb firstly look for the property mapguide.pathTo.webconfig.ini inside  
configuration.properties. Secundarily, if missing, Geoweb looks for the file  
webconfig.ini under static contents folder.As last chance Geoweb looks for  
the file webconfig.ini under the classPath.  
Property mapguide.pathTo.webconfig.ini had been found inside  
configuration.properties file with valorized  
S:/deploy/GW_MODULES_MG31/conf/webconfig.ini  
Configured path refers to an existing file  
Initializing MapGuide with the file:  
S:/deploy/GW_MODULES_MG31/conf/webconfig.ini  
applyResourcePackageInAllMapGuideServer()  
Checking if there is a resource package to apply to MapGuide Server..  
Looking for folder S:\deploy\GW_MODULES_MG31\WEB\geowebfolder\mapguide..  
Folder S:\deploy\GW_MODULES_MG31\WEB\geowebfolder\mapguide exists  
building all the resource package in folder:  
S:\deploy\GW_MODULES_MG31\WEB\geowebfolder\mapguide  
building resource package named: cde_test_2.mgp  
siteAddressList: [192.168.0.225, 192.168.0.168]  
applyResourcePackage() - resourcePackage:  
S:\deploy\GW_MODULES_MG31\WEB\geowebfolder\mapguide\cde_test_2.mgp  
computeMd5Hex() - file:  
S:\deploy\GW_MODULES_MG31\WEB\geowebfolder\mapguide\cde_test_2.mgp  
md5 computed in nanoseconds: 12289400  
computed md5Hex: 19f5ddb761d9bb4dc3668e5a1df234df  
  * nAddressLeft: 2  
  * attemp number: 1  
  * currentSiteAddress: 192.168.0.225  
  * Resource package correctly applied to MapGuide site address:  
192.168.0.225  
  * nAddressLeft: 1  
  * Resource package had not been updated on n 1 MapGuide address:  
[192.168.0.168]  
done  
-----  
-----
```

Proprietà 'geowebfolder' configurata ma incorretta. Oppure mancanza della folder 'mapguide' dentro la geowebfolder

```
-----  
-----  
15-10-2019 10:21:18,732 [localhost-startStop-1] GEOWEB INFO  
  
MgInizializer - init()
```



```

Now Geoweb wil try to load the webconfig.ini. This is the loading priority.
Geoweb firstly look for the property mapguide.pathTo.webconfig.ini inside
configuration.properties. Secondly, if missing, Geoweb looks for the file
webconfig.ini under static contents folder.As last chance Geoweb looks for
the file webconfig.ini under the classPath.
Property mapguide.pathTo.webconfig.ini had been found inside
configuration.properties file with valorized
S:/deploy/GW_MODULES_MG31/conf/webconfig.ini
Configured path refers to an existing file
Initializing MapGuide with the file:
S:/deploy/GW_MODULES_MG31/conf/webconfig.ini
applyResourcePackageInAllMapGuideServer()
Checking if there is a resource package to apply to MapGuide Server..
Looking for folder S:\deploy\GW_MODULES_MG31\WEB\geowebfolder\mapguide..
Cannot load any resource package. Property geowebfolder inside file
configuration.properties results configured but is incorrect.Resource
package file must be placed under the specified folder, respecting the path:
[geowebfolder]/mapguide/ and named with extension: .mpg.Please check that
the configured name match the folder name in file system.
done
-----
-----

```

Due IpAddress configurati (uno esistente ed uno finto); un file .mpg trovato; elaborazione .mpg fallita perchè file aperto con un programma esterno (7zip) durante lo startup del server; risorsa caricata solo su un server (ma senza elaborazione);

```

-----
-----
15-10-2019 10:04:34,395 [localhost-startStop-1] GEOWEB INFO

MgInizializer - init()
Now Geoweb wil try to load the webconfig.ini. This is the loading priority.
Geoweb firstly look for the property mapguide.pathTo.webconfig.ini inside
configuration.properties. Secondly, if missing, Geoweb looks for the file
webconfig.ini under static contents folder.As last chance Geoweb looks for
the file webconfig.ini under the classPath.
Property mapguide.pathTo.webconfig.ini had been found inside
configuration.properties file with valorized
S:/deploy/GW_MODULES_MG31/conf/webconfig.ini
Configured path refers to an existing file
Initializing MapGuide with the file:
S:/deploy/GW_MODULES_MG31/conf/webconfig.ini
applyResourcePackageInAllMapGuideServer()
Checking if there is a resource package to apply to MapGuide Server..
Looking for folder S:\deploy\GW_MODULES_MG31\WEB\geowebfolder\mapguide..
Folder S:\deploy\GW_MODULES_MG31\WEB\geowebfolder\mapguide exists
building all the resource package in folder:
S:\deploy\GW_MODULES_MG31\WEB\geowebfolder\mapguide
building resource package named: cde_test_2.mpg
S:\deploy\GW_MODULES_MG31\WEB\geowebfolder\mapguide\cde_test_2.mpg:

```

```
Impossibile accedere al file. Il file è utilizzato da un altro processo.
Error building resource package named: cde_test_2.mgp
siteAddressList: [192.168.0.225, 192.168.0.168]
applyResourcePackage() - resourcePackage:
S:\deploy\GW_MODULES_MG31\WEB\geowebfolder\mapguide\cde_test_2.mgp
computeMd5Hex() - file:
S:\deploy\GW_MODULES_MG31\WEB\geowebfolder\mapguide\cde_test_2.mgp
md5 computed in nanoseconds: 6628700
computed md5Hex: 1f769acff2fc69590a87699c4fbafe7f
* nAddressLeft: 2
* attemp number: 1
* currentSiteAddress: 192.168.0.225
* Resource package correctly applied to MapGuide site address:
192.168.0.225
* nAddressLeft: 1
* Resource package had not been updated on n 1 MapGuide address:
[192.168.0.168]
done
-----
-----
```

Workflow

WorkflowInizializerService

WorkflowInizializerService, as well as common xInizializerService stuff, performs an automatic Activiti Process Definition (.bpmn file) deploy.

Before **gw version 4.4.4** it was possible to handle Activiti deployments only from the Webadmin (manually managing the .bpmn files) Now there is an automatic procedure that look for all the Process Definitions files with .bpmn extension existing under this path

```
[geowebfolder]/activiti/
```

The [geowebfolder] is configurable under configuration.properties file. If the file is found, GeoWeb try to load to all the available .bpmn files and to deploy them inside Activiti workflow engine.

In geoweb exist the following convention:

Activiti Process Definition ID \Leftarrow matches \Rightarrow GwProcess name

This convention is respected even here: the name of every .bpmn file HAVE TO MATCH with one GwProcess name. If a relation is found, the .bpmn is deployed and linked to the GwProcess. This is not an Activiti requirement, that blocks the ProcessDefinition deploy, but it is simply an our choice in aim to avoid to erroneously deploying things unrelated to GwProcess.

There is a mechanism that ensures that a single .bpmn file, if not modified subsequently, it will not to be deployed again on Activiti on the next server startups: a md5 check is performed at this purpose. If

we want to exclude that md5 check there is a flag to enable inside configuration.properties (**default false**)

```
workflow.activiti.forceDeploy.enabled=true
```

Personalizzazioni Loghi

Dalla versione 4.4.18 di Geoweb Framework è stata introdotto la possibilità di personalizzare lo sfondo dell'home page del framework (indexBackground), il logo sulla login di geoweb e i banner left, center e right a livello globale.

Per farlo configurare nel configuration.properties i seguenti parametri:

- Personalizzazione dello **sfondo** in Home Page: inserire nel configuration.properties il parametro **images.IndexBackground** a cui andrà assegnata una stringa corrispondente al nome dell'immagine scelta come sfondo, compresa di estensione. L'immagine dovrà poi essere salvata nei contenuti statici sotto la cartella WEB

```
images.IndexBackground=MyCustomIndexBackground.jpg
```

Assicurarsi di avere la seguente configurazione nel file spring-security.xml:

```
<intercept-url pattern="/resources/#{  
T(com.geowebframework.metadataaservice.registry.GwReleaseInfos).getRevision()  
}/images/${images.login.form.logo:gw4_enterprise.png}" access="permitAll" />
```

- Personalizzazione del **logo** nella login: inserire nel configuration.properties il parametro **images.login.form.logo** a cui andrà assegnata una stringa corrispondente al nome dell'immagine scelta come logo, compresa di estensione. L'immagine dovrà poi essere salvata nei contenuti statici sotto la cartella WEB\images

```
images.login.form.logo=MyCustomLoginLogo.png
```

Assicurarsi di avere la seguente configurazione nel file spring-security.xml:

```
<intercept-url pattern="/resources/#{  
T(com.geowebframework.metadataaservice.registry.GwReleaseInfos).getRevision()  
}/${images.IndexBackground:IndexBackground.jpg}" access="permitAll" />
```

- Personalizzazione dei **banner**: inserire nel configuration.properties il parametro **images.bannerLeft** per la personalizzazione del banner di sinistra, **images.bannerCenter** per la personalizzazione del banner centrale e **images.bannerRight** per la personalizzazione del banner a destra. Ad ogni parametro andrà assegnata una stringa corrispondente al nome dell'immagine scelta come banner, compresa di estensione. L'immagine dovrà poi essere salvata nei contenuti statici sotto la cartella WEB. Se presenti i parametri nel configuration.properties, queste configurazioni andranno a sostituire globalmente le singole personalizzazioni settate per progetto con i consueti [tag nell'editor xml](#) di progetto

```
images.bannerLeft=MyCustomBannerLeft.jpg  
images.bannerCenter=MyCustomBannerCenter.jpg
```

Last
update:
2022/03/30 gwusermanual:configuration_4_4_x https://wiki.geowebframework.com/doku.php?id=gwusermanual:configuration_4_4_x&rev=1648651923
16:52

images.bannerRight=MyCustomBannerRight.jpg

From:
<https://wiki.geowebframework.com/> - **GeowebFramework**

Permanent link:
https://wiki.geowebframework.com/doku.php?id=gwusermanual:configuration_4_4_x&rev=1648651923

Last update: **2022/03/30 16:52**

