Cegid XRP Ultimate

Workflow Information Manager

cegid

Document de release de la version H2.01

Ce document présente les évolutions survenues sur le module Qualiac® Workflow Information Manager en H2.01.

Comment garder l'iso fonctionnalité

Au niveau des WebServices, l'objet QualiacAuthenticator reçoit une propriété (facultative) supplémentaire : la notion de timestamp. Cela n'a pas d'impact sur les implémentations que nous avons testées (en java, C# et ActionScript), mais si vous rencontrez des difficultés, il sera peut-être nécessaire de re-générer les stubs.

Fonctionnalités

Nouveautés

Utilisation directe des paramètres dans une mise en forme

Explications

Jusqu'à la version H1.01 comprise, quand on voulait passer un paramètre issu d'un critère de traitement (cas WIM par job) ou un paramètre passé dans l'URL (cas WimServlet), il fallait passer par une variable intermédiaire (alias).

Depuis la H2.01, il est possible de sauter cette étape en utilisant directement dans la mise en forme le tag $$CP_xxxx avec xxxx = nom du paramètre.

Exemple : en utilisant le tag $$CP_PARAM01 dans la mise en forme, et lors d'un appel de WimServlet du style : ...action=select&requete=TEST&$PARAM01=Test , on va retrouver la chaîne de caractères "Test" en lieu et place de $$CP_PARAM01.

Ce principe est le même dans le cas d'un WIM lancé par job si on veut utiliser la valeur d'un paramètre de job. Par exemple, en utilisant le tag $$CP_CHAINE9D, on pourra afficher le contenu du paramètre de job CHAINE9D.

WebService : Expiration du jeton d'authentification

Explications

Une nouvelle notion peut être gérée dans le jeton d'authentification : la notion de durée de validité.

Pour cela, ajouter la ligne
WEBSERVICE.authExpires=xxxx (avec xxxx = durée de vie du token en milliseconde)
dans le fichier qualiacdb.properties.
Dans ce cas, le système générera un jeton dépendant du timestamp. Une fois le délai xxxx dépassé, le jeton ne sera plus utilisable (retour d'une erreur ERROR WS-0036), il faudra alors en redemander un nouveau.
A noter qu'une nouvelle méthode utilitaire remainsToken est disponible. Elle prend en entrée le timestamp du jeton et retourne la durée de validité restante pour le jeton, ou -1 si le jeton a déjà expiré.

WebService : Tracer les messages Soap entrant et sortant

Explications

En positionnant la valeur à "true" dans la ligne :
WEBSERVICE.AllSoapLog=true
Tous les messages SOAP seront tracés (écriture dans la console "System.out").

WebService : Code erreur Http sur mauvaise authentification

Explications

Dans le fichier qualiacdb.properties, en positionnant une valeur pour la ligne :

WEBSERVICE.AuthHttpError=XXX (avec XXX = code erreur HTTP, typiquement 401 dans notre cas)
le WebService renverra ce type d'erreur dans le cas d'une erreur d'authentification.

WebService : mode d'authentification pour comptes techniques

Explications

Dans le fichier qualiacdb.properties, en positionnant la ligne :
WEBSERVICE.toggleCnxMethod=true
et si la configuration pour l'authentification est en mode LDAP et que celle-ci échoue, on tente, dans la foulée, une authentification en mode "base". Cela peut permettre d'éviter d'avoir à définir des comptes techniques dans l'active directory.

Wim : nouveau symbole $$SENDER

Explications

Dans une mise en forme WIM, il est possible d'utiliser un nouveau symbole $$SENDER. Ce dernier sera remplacé par l'adresse mail définie comme émetteur du message.

GKTUMEL : Envoi de mail depuis Qualiac

Une nouvelle fonctionnalité est accessible depuis n'importe quelle gestion de Qualiac® : l'envoi de mail. Il est possible de définir les destinataires (et leur type : principal, copie, copie cachée), le texte qui va être envoyé, les pièces à joindre, etc.

Explications

Cela se fait au travers de la gestion des messages électroniques (GKTUMEL), accessible uniquement en synchronisation d'une autre gestion.

Pour la mise en place, voici ce qui doit être réalisé :

- Dans un premier temps, il est nécessaire de procéder au paramétrage des messages électroniques (GTUPME) et de définir, pour l'application et l'objet, les grandes lignes qui vont contrôler le comportement de l'envoi des mails. On peut notamment préciser qui sera l'émetteur du mail, à qui sera adressé la réponse quand le destinataire fera "répondre", s'il faut mettre des personnes en copie, comment seront regroupées les envois (un seul mail, plusieurs mails regroupés par domaine, autant de mails que de destinataires, etc.). Enfin il est possible de coupler avec une requête WIM qui sera exécutée et dont la mise en forme sera appliquée pour compléter le contenu du mail. A noter : selon l'application d'origine (e-Achats, XRM, ...) le contenu des menus déroulants est variable.

- Définir une synchronisation entre la gestion appelante et la gestion des messages (GKTUMEL). Il faut donc lier les mnémoniques (via GAMN) et donner les paramètres de synchronisation passés en entrée de GKTUMEL.

Exemple : Mise en place depuis l'écran des gestionnaires (GGES).
- Associer GGES et GKTUMEL dans GAMN
- Définir une configuration dans GTUPME, pour l'objet "oeiges", l'application ODE
- Définir les paramètres ENTITE (préciser comme constante 'OEGES') et IDENT1 (passer la colonne NUMOEGES) en sortie dans GTFPS pour la gestion OEIGES.
Une fois ceci terminé, il sera possible d'accéder à la gestion GKTUMEL afin de saisir le texte, compléter les options, ajouter des destinataires et/ou des pièces jointes.

Il est possible d'associer une requête WIM pour gérer la mise en forme du mail envoyé. Pour cela, il suffit d'avoir préalablement défini cette requête dans GTUREQ de façon normale (comme une sous-requête classique) et d'utiliser, dans la mise en forme l'alias réservé $INNER_TXTTUMEL. Lors de l'exécution, celui-ci sera remplacé par le texte saisi dans GKTUMEL.

Modifications

WimServlet et WebServices : lecture de la GED

Explications

Ajout de nouveaux paramètres en entrée de l'action getdoc :

- doc_like : lors de la recherche avec l'identifiant précisé dans le paramètre p01, si doc_like vaut O on ne fait pas une recherche stricte mais on fait "like".

Exemple, avec p01=ETS%&doc_like=O, on récupérera la GED pour les identifiants qui commencent par ETS

- doc_listFile : si pour une raison ou pour une autre, on connait déjà la liste des fichiers que l'on veut ramener, on peut la passer ici (sous forme de nom de fichiers séparés par des ;)

Exemple : doc_litsFile=plan.pdf;doc.png

- doc_listExt : seuls les fichiers portant sur la liste d'extension passés dans ce paramètre seront ramenés.

exemple : doc_listFile=pdf;xls

- doc_listDir : seuls les fichiers stockées dans les répertoires passés dans ce paramètre seront ramenés.
- doc_dom : certaines applications Qualiac® ont ajouté une surcouche quant à la façon de gérer la GED. C'est notamment le cas pour Qualiac® e-Achats. Si on veut accéder à la GED de cette application, il faudra donc préciser pi_dom=0 (zéro).

Enfin, lors de la résolution des noms et/ou des chemins d'accès aux fichiers, il était déjà possible d'utiliser des variables dans le paramétrage des types de documents GTDTY ($P01, $TYP, $DEN ...). Il est aussi possible d'utiliser des symboles ($ETS, $USER ...), mais pour résoudre ces derniers nous avons besoin de quelques informations plus techniques.

Donc, si des symboles sont utilisés, il faudra passer en supplément 5 paramètres :

- $$DOC_SYMBETS : établissement
- $$DOC_SYMBDATE : date
- $$DOC_SYMBUSER : utilisateur
- $$DOC_SYMBUSERW : utilisateur Window
- $$DOC_SYMBWSA : poste de travail