

3.19 Envoi de courriers électroniques



Bon pour relecture

Le service d'envoi de courriers électroniques a été initialement développé pour le gestionnaire d'exceptions (certaines implémentations permettent d'envoyer les rapports d'exception par courrier électronique). Ce service est disponible à toute autre fin (envoi d'informations, remontée d'alertes, ...).

Sommaire :

- [Implémentations](#)
- [Configuration](#)
- [Utilisation](#)

Implémentations

esup-commons prévoit plusieurs implémentations du service d'envoi de mails :

- **SimpleSmtplibServiceImpl** permet l'envoi simple de mails.
- **AsynchronousSmtplibServiceImpl** permet l'envoi de mails à l'aide d'un fil d'exécution spécifique de priorité basse. Cette fonctionnalité évite les timeout observés lors de l'expédition d'un grand nombre de mails.

C'est cette dernière implémentation qui est généralement conseillée.

Configuration

Le service d'envoi de mail se configure à l'aide du fichier `/properties/smtp/smtp.xml`.

Exemple de déclaration :

```
<bean
    id="smtpService"
    class="org.esupportail.commons.services.smtp.AsynchronousSmtplibServiceImpl"
    lazy-init="true"
    >
    <property name="servers">
        <list>
            <ref bean="smtpServer" />
        </list>
    </property>
    <property name="fromAddress" ref="smtpFromAddress" />
    <property name="interceptAll" value="{smtp.interceptAll}" />
    <property name="interceptAddress" ref="smtpIntercept" />
    <property name="notInterceptedAddresses" value="{smtp.notInterceptedAddresses}" />
    <property name="testAddress" ref="smtpTestAddress" />
    <property name="charset" value="{smtp.charset}" />
</bean>
```

Propriétés :

- **Servers** est une liste de serveurs à utiliser pour envoyer les messages. Les serveurs sont utilisés suivant leur ordre d'apparition dans le fichier. Si, et seulement si, une exception est levée lors de l'envoi du mail sur un serveur le suivant est utilisé.
- **fromAddress** pointe vers un *bean* définissant l'expéditeur des messages.
- **interceptAll** booléen **true/false** définissant si on intercepte ou non l'ensemble des messages émis par l'application
- **interceptAddress** pointe vers un *bean* définissant le destinataire des messages en phase de mise au point. Tous les messages sont envoyés à cette adresse afin d'éviter un effet de *spam* (y compris les exceptions, la propriété **recipientEmail** du fichier `/properties/exceptionHandling/exceptionHandling-example.xml` n'étant pas prise en compte). Il suffira de positionner **interceptAll=true** pour retrouver un comportement normal.
- **notInterceptedAddresses** liste des adresses (en minuscule et séparé par une virgule) pour lesquelles les e-mails envoyés ne seront pas interceptés.
- **testAddress** pointe vers un *bean* définissant le destinataire du message généré par la méthode `test()` qui peut être utilisée dans les tests unitaires par exemple.

Les *beans* **smtpFromAddress**, **smtpInterceptAddress** et **testAddress** sont tous de type `javax.mail.internet.InternetAddress`.

Exemple de déclaration :

```
<bean
  id="smtpFromAddress"
  class="javax.mail.internet.InternetAddress">
  <property name="address" value="webmaster@domain.org" />
  <property name="personal" value="WebMaster" />
</bean>
```

Propriétés :

- **address** est l'adresse email.
- **Personal** est le nom de l'expéditeur qui apparaîtra dans le client de messagerie du destinataire.

Les *beans* listés dans la propriété **Servers** de **smtpService** sont tous de type **org.esupportail.commons.services.smtp.SmtpServer**.

Exemple de déclaration :

```
<bean
  id="smtpServer1"
  class="org.esupportail.commons.services.smtp.SmtpServer">
  <property name="host" value="smtp.domain.org"/>
  <property name="port" value="25"/>
  <property name="user" value="" />
  <property name="password" value="" />
</bean>
```

Propriétés :

- **host** est l'adresse internet du serveur SMTP
- **port** est le port du service SMTP sur la machine **host**
- **user** est le nom de l'utilisateur utilisé pour envoyer les messages. Si cette propriété est vide ou commentée la connexion au service SMTP se fait en anonyme.
- **password** est le mot de passe lié à **user**. Cette propriété est ignorée en cas de connexion anonyme.

Utilisation

Pour qu'un de vos *beans* puisse utiliser le service d'envoi de mails il faut qu'il ait une référence, définie dans le fichier de configuration *Spring*, sur un **smtp Service**.

Ensuite il peut utiliser ce service, par exemple :

```
smtpService.send(to, emailSubject, htmlBody, textBody);
```

- **to** est le destinataire (un **InternetAddress**).
- **emailSubject** est le sujet du mail.
- **htmlBody** et **textBody** sont, respectivement, les contenus HTML et texte du message. Au moins un de ces deux paramètres doit être non **null**. Si les deux sont renseignés, le mail généré est alors de type **MimeMultipart**.