Почтовый шлюз

Системе CM 4/5 необходима возможность приёма документов по каналам электронной почты (e-mail). Обычно для этой цели создаётся специальный почтовый ящик с известным контрагентам именем (адресом). Приходящие на этот адрес сообщения обрабатываются системой, если удовлетворяют некоторым условиям (правилам).

Реализация почтового сервера в рамках платформы ECMp5 в настоящее время не планируется. Поэтому для организации почтового шлюза необходим внешний почтовый сервер, на котором его администратором должен быть создан один или несколько почтовых ящиков, предназначенных для приёма корреспонденции для системы. Система должна периодически опрашивать этот(эти) ящик(и), забирая из него(них) пришедшие сообщения.

Взаимодействие системы с этим почтовым сервером должно осуществляться по одному из широко распространённых протоколов POP3 или IMAP. JDK и Spring имеют удобные средства, использующие эти протоколы (Java Mail и Spring Mail, соответственно).

Следует отметить, что задача приёма сообщений по e-mail (документов, запросов, управляющих команд и др.) является достаточно общей для различных систем. Поэтому сервис получения сообщений может быть реализован в рамках платформы ECMp5. Принятые сообщения могут быть помещены в доменные объекты. Для этого понадобится создать в системной конфигурации отдельный тип (E-mail message).

В отличие от процедуры взаимодействия с почтовым сервером, обработка пришедших сообщений специфична для каждой системы. В некоторых системах сортировка и обработка e-mail будет проводиться вручную специальным пользователем, в других – частично или полностью автоматически, по соответствию формальным условиям. В первом варианте разработчики системы должны будут создать одну или несколько специальных страниц в GUI, отображающих доменные объекты сообщений e-mail, а также специфические для них действия (если необходимо). Для второго варианта хорошим способом реализации является создание специального бизнес-процесса (Activiti). Связывание этого процесса с компонентом приёма сообщений может осуществляться посредством конфигурации.

# Компоненты платформы

## Периодическая задача – приём сообщений

Каждый экземпляр этой задачи собирает сообщения из одного почтового ящика, поэтому в настройках задачи должны присутствовать имя пользователя (ящика) почтового сервиса и пароль для выполнения аутентификации на этом сервере (желательно, обратимо зашифрованный). Возможно добавление необязательного параметра, хранящего адрес (имя) почтового сервера, что позволит организовать сбор почтовых сообщений с ящиков, расположенных на разных серверах (в разных доменах).

Задача при каждом запуске соединяется с почтовым сервером, аутентифицируется под заданным пользователем и запрашивает все сообщения, хранящиеся на сервере для данного пользователя (в данном ящике). Каждое полученное сообщение сохраняется в новый доменный объект, после чего (только при успешном сохранении!) удаляется с почтового сервера. Особо следует отметить необходимость сохранения каждого сообщения в отдельной не вложенной транзакции, чтобы при ошибке сохранения одного сообщения предыдущие, уже удалённые с сервера, не были потеряны.

Ещё одним важным параметром настройки задачи является имя процесса, запускаемого для каждого успешного принятого и сохранённого сообщения. Однако, этот параметр не должен быть обязательным, т.к. автоматическая обработка в конкретной системе может и не производиться.

## Конфигурация

Конфигурация платформы для почтового шлюза должна содержать тип доменного объекта – e-mail message. Помимо полей для хранения заголовков и содержания сообщения, объект должен позволять сохранять вложения (вложенные файлы).

Также в глобальных настройках должен быть сконфигурирован адрес (или доменное имя) почтового сервера. Даже в случае реализации аналогичного параметра в настройках задачи приёма сообщений глобальная настройка полезна для задания значения по умолчанию.

# СПЯ CompanyMedia 4/5

## Процесс обработки сообщений e-mail

В соответствие с логикой более ранних версий CompanyMedia, бизнес-процесс обработки входящих сообщений должен производить сопоставление адреса отправителя сообщения со справочником организаций, в случае успеха связывать сообщение с найденной организацией и назначать ответственному за эту организацию пользователю задачу (UserTask) по обработке данного сообщения. При выполнении этой задачи пользователь может выбрать создать из сообщения документ (входящий или ОГ) либо проигнорировать/отвергнуть сообщение (в этом случае процесс завершается).

## Компоненты процесса

Поскольку описанный выше процесс предполагает довольно сложную логику как при поиске/связывании организации, так и при формировании объекта документа, оба эти шага процесса представляется разумным реализовывать на Java в виде ServiceTasks.

## Действия

Для документов, создаваемых пользователем вручную по сообщению, может понадобиться специальное действие, выполняющее предзаполнение полей документа перед предоставлением его пользователю для редактирования. Таких действий может быть несколько, если по e-mail могут приниматься документы разных типов.