Projet ROSES

Programme MDCO – Edition 2007

Livrable no D1.1
Fonctionnalités d'un système ROSES

Identification

Acronyme du projet	ROSES
Numéro d'identification de l'acte attributif	ANR-07-MDCO-011-01
Coordonnateur	Paris 6
Rédacteur (nom, téléphone, email)	Bernd Amann 01 44 27 70 09 Bernd.Amann@lip6
No. et titre	D1.1 Fonctionnalités d'un système ROSES
Version	v0.1
Date de livraison prévue	30 juin 2008 / t0+6
Date de livraison	février 2009 / t0+12

Résumé

Ce document décrit les différentes fonctionnalités d'un système ROSES. Ces fonctionnalités sont génériques et indépendantes d'une architecture technique particulière (voir livrable D1.2) où d'une application précise.

Table des matières

A Introduction

Ce document décrit les différentes fonctionnalités de syndication et d'aggrégation d'un système ROSES. Ces fonctionnalités sont génériques et définies indépendamment d'une architecture technique (voir livrable D1.2) où d'une application particulière. On considère qu'il est possible de construire des systèmes ROSES qui choisissent et adaptent les fonctionnalités présentées aux contraintes et besoins de différents cas d'usage :

Diffusion et partage : ROSES sert comme moyen de diffusion et de partage ciblé d'informations. Les publications et les souscriptions aux flux ROSES reflètent des liens sociaux entre les personnes. ROSES permet en particulier aux utilisateurs de protéger et de mieux contrôler leurs liens sociaux avec d'autres utilisateurs et d'autres informations personnelles (architecture distribuée) pour créer des espaces de diffusion et de partage de type « social bookmarking » (Del.icio.us) et « blogging/microblogging » (Blastfeed, Twitter).

Surveillance et détection d'événements : La surveillance consiste à combiner et analyser des collections de flux RSS pour la détection rapide d'événements . Un type d'application concerne la surveillance des cours de la bourse (application traditionnelle flux de données), mais également la surveillance détection d'événements plus complexes qui intègrent des flux RSS complémentaires.

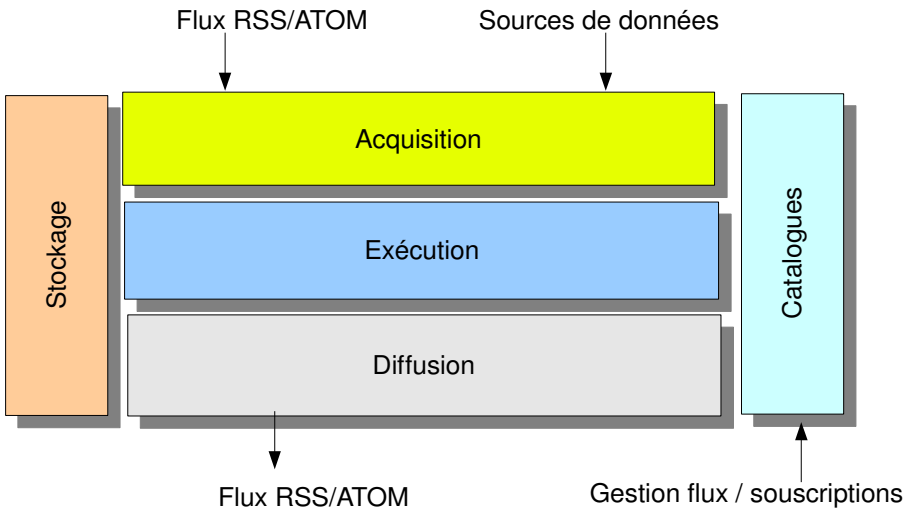
Valorisation : Les flux sont combinés avec des sources (bases de données, ontologies) pour enrichir (jointures sémantiques/textuelles) l'information diffusée. Un exemple d'application est la surveillance et l'archivage de sources médicales/scientifiques sur une maladie.

Personnalisation : Le système sert à créer et maintenir dynamiquement un espace d'information personnalisé qui combine des flux externes avec des données personnelles (agenda, carnet d'adresses, mail) et des préférences utilisateurs. Les publications sont à priori “privées” (usage personnel). Un exemple d'application est la génération d'un journal personnalisé (Mon Monde).

Archivage : Les flux RSS correspondent en général à des informations « éphémères » (non-persistantes) et la plupart des applications précédentes nécessitent « a priori » pas le stockage de flux à long terme. Néanmoins, les flux RSS représentent de plus en plus une source d'information pour des applications qui mettent en oeuvre des techniques d'« apprentissage » et de « fouille de données » et qui nécessitent l'accès à des archives de flux volumineux.

B Couches fonctionnelles

Les différentes fonctionnalités principales d'un système ROSES sont illustrées dans la Figure 1. Elles sont structurées en trois couches superposées d'acquisition, d'exécution et de diffusion de flux RSS assistées par une couche contrôle pour la gestion des flux et des souscriptions et une couche stockage pour le stockage à court (cache) et à long (archivage) terme de données.



1. Couches fonctionnelles principales d'un système ROSES

1. Couche « Acquisition »

Cette couche est responsable de l'acquisition des données externes (souscriptions) et de leur transformation en flux ROSES (voir livrable D2.2 Modèle de données). Elle est capable de gérer flux RSS/ATOM mais également d'autres types de sources de données (bases de données, services Web, capteurs) accessibles à travers une interface XML et un protocole sans (HTTP) ou avec état (REST, SOAP).

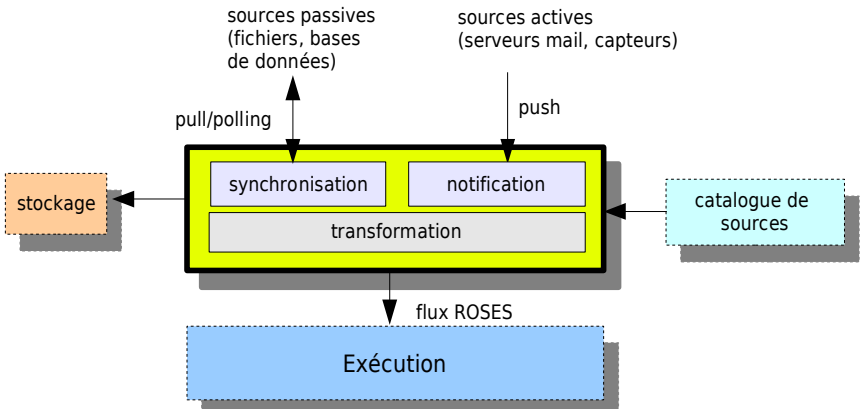


Illustration 1: Module "Acquisition"

On distingue entre les sources *passives* qui doivent interrogées (mode pull) et les sources *actives* qui génèrent des notifications en mode subscribe/notify (mode push). Le module Acquisition remplit trois fonctionnalités principales (Illustration 1: Module "Acquisition") :

1. synchronisation de sources passives (voir livrable D2.4).
2. gestion de notifications
3. transformation de données

La distinction entre source active et source passive est indépendantes de la distinction entre *flux de données* et *sources de données statiques* (fichiers/bases de données). En particulier, les flux RSS sont des *flux de données* publiés sous forme de *sources passives* qui doivent être interrogées périodiquement (polling) pour détecter l'apparition de nouveaux items. L'accès aux bases de données s'effectue généralement à travers un service web qui encapsule une requête. Ce service transforme les données externes en données internes conformes au modèle ROSES. La transformation d'une source enregistrée consiste à générer un flux d'items ROSES qui déclenche l'activation des souscriptions (voir couche de traitement). Le système implante par défaut une fonction de transformation d'items RSS/ATOM en item ROSES. D'autres types de sources nécessitent la définition de fonctions de transformation spécifiques à la structure de données reçues.

Formats d'acquisition :

RSS / ATOM : formats de syndication standards
ROSES : format XML pour l'échange de flux ROSES entre les pairs d'un système ROSES distribuées.
autres formats : les données sont reçues en format XML sans restriction sur leur structure (DTD); chaque format nécessite la définition d'un transformateur/wrappers adapté.

Protocoles d'acquisition : il n'y a pas de restriction particulière sur les protocoles utilisés pour l'acquisition (et la diffusion) des données :

HTTP(S) : protocole simple (sans état) pour l'accès en mode pull (get) et push (put)
REST / SOAP / XML-RPC
SMTP
WSP (protocole ROSES)

2. Couche «Exécution»

Une publication ROSES est une *requête continue* qui observe et interroge un ou plusieurs flux d'entrée. La couche Exécution détecte et exécute les publications déclenchées par les flux ROSES (Voir Illustration 2: Couche Exécution). Toutes les publications sont enregistrées dans le catalogue de publications.

La détection de publication consiste

1. à identifier efficacement les publications (requêtes continues) concernées par les événements des flux ROSES,
2. à transmettre (si nécessaire) les nouveaux items reçus à la couche stockage,
3. à envoyer les publications détectées au module d'exécution.

L'exécution d'une publication consiste dans l'instantiation et l'évaluation d'une requête ROSES (voir livrable *D2.2* RSS-XML model and algebra) qui interroge (si nécessaire) les données stockées dans le module de stockage. Le résultat est un flux ROSES nommé transmis au module de diffusion.

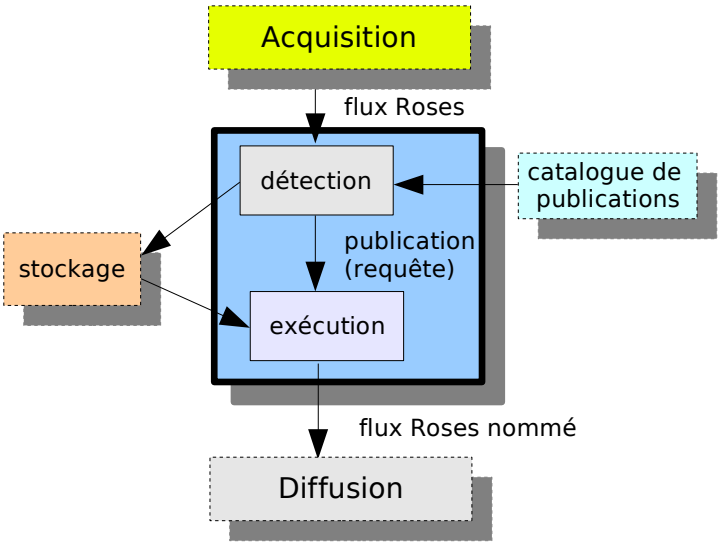


Illustration 2: Couche Exécution

3. Couche « Diffusion »

La couche de diffusion est responsable du formatage et de la diffusion des souscriptions. Elle prend en entrée un flux ROSES nommé (publication) et identifie l'ensemble des souscriptions abonnées au flux. Dépendant du format de diffusion associé à chaque souscription, elle transforme les items ROSES en item RSS ou ATOM, messages SOAP, messages mail, etc... Il existe également une syntaxe XML pour la diffusion d'items ROSES sans transformation.

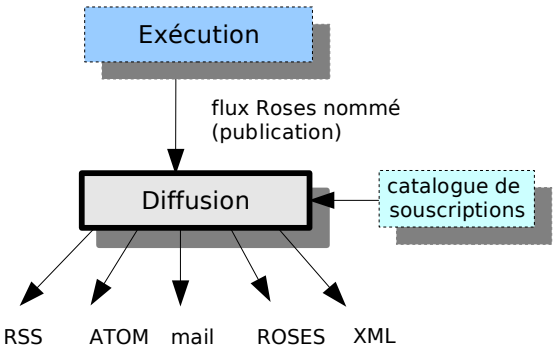


Illustration 3: Couche Diffusion

Formats de diffusion :

RSS / ATOM : format de syndication standard pris en compte par une multitude de lecteurs

de flux;

ROSES : format d'échange de flux entre pairs ROSES

XML

Protocoles de diffusion :

HTTP
WSP : protocole pour la synchronisation de flux ROSES (pull/push)
SMTP : protocole pour l'envoi de messages (push)
SOAP / REST / XML-RPC

C Contrôle

1. Catalogue de sources (flux externes)

La couche contrôle propose un langage de haut niveau pour enregistrer les différents types de ressources dans le système. L'enregistrement d'une source consiste à spécifier son adresse (URL), un protocole de communication, ainsi que d'autres paramètres qui dépendent du type de la source (structure de données, paramètres d'interrogation).

Le système fournit par défaut une fonction d'enregistrement de flux RSS/ATOM et de flux ROSES. L'enregistrement de bases de données ou de services Web nécessite la définition de wrappers spécifiques qui transforment des résultats de requêtes ou d'appels de services en flux ROSES. Le choix des fonction de transformation est contrôlé par la notion de type de sources qui correspond à un type abstrait de donnée avec une fonction de transformation en flux ROSES. La fonction de transformation peut être paramétré pour, par exemple, filtrer des données avant la transformation en items ROSES.

Données d'enregistrement

information de connexion : adresse (URL), protocole
gestion : temps d'expiration et temps de validité
statistiques : fréquences de mises-à-jour, bande passante
transformation : type de données

Services

création d'un enregistrement de sources
effacement d'un enregistrement
recherche d'un enregistrement

Langage

Enregistrement d'une source !

```
 register source <name> :            : nom du flux ROSES généré
  <URL>                              : URL de la source
  of type <type source>              : type de la source (RSS, ATOM, ROSES,...)
                                       (optionnel; RSS)
  update <frequency>                 : fréquence de rafraîchissement
                                       (optionnel; par défaut automatique)
  until <date>                       : date de fin de rafraîchissement
                                       (optionnel; par défaut forever)
  expires <date>                     : date d'expiration de l'enregistrement
                                       (les données peuvent être effacés)
                                       (optionnel : par défaut immediately)
   keywords <mots clés>              : séquence de mots clés qui décrit le flux
                                       enregistré (optionnel)
```

Effacement d'un enregistrement :

```
delete source <name>
```

Recherche / affichage d'un enregistrement :

```
show source <name>
show sources of type <type>
show sources with keywords <mots clés>
```

Exemples

Enregistrement d'un flux RSS généré par des alertes Google :

```
register source googlealerteuro2008 :
   http://www.google.com/alerts/feeds/05544618421063530945/18306496925904217652
   until september 2008 keywords euro, football
```

Enregistrement d'une page web généré par une requête Google (nécessite la définition d'un type googlepage qui transforme une page de réponse Google en flux ROSES) :

```
register source googlesearcheuro2008 :
   http://www.google.fr/search?q=euro+2008&sourceid=navclient-ff&ie=UTF-
   8&rls=GGGL,GGGL:2006-35,GGGL:fr of type googlepage update daily until
   september 2008 expires december 2008 keywords euro, football
```

Enregistrement d'une page web (nécessite la définition d'un type wikipage) :

```
register source wpuefa : http://en.wikipedia.org/wiki/Uefa of type wikipage
   update once keywords football
```

Enregistrement d'un flux RSS du site UEFA :

```
register source uefarrs : http://www.euro2008.uefa.com/rss/index.xml update
   hourly keywords football
```

Enregistrement d'un document XML avec la liste des joueurs favoris :

```
register source myplayers : c:\MyPlayers.xml//player update daily keywords
```

```
   football
```

- Enregistrement (intensionnel) d'une page par joueur dans le feed myplayers :

```
for $p in feed(myplayers) register source $p_name :
   http://en.wikipedia.org/wiki/Uefa/($p/name) update once keywords football,
   $p/name
```

2. Catalogue de publications (flux internes)

Une publication est une requête continue nommée sur les sources enregistrés (flux externes) et les flux publiés (flux internes).

Données de publication

- nom
- utilisateur
- requête ROSES
- date d'expiration
- mots clés

Services

- enregistrement de publication
- effacement de publications
- recherche de publications

Langage

```
publish feed <name> :                  : nom du flux publié
         <requête>                     : requête
         [expires <date>]              : date d'expiration de la publication
         [keywords <mots clés>]        : séquence de mots clés qui décrit le flux
                                         enregistré
```

```
delete feed <name>
```

Exemples

- News dans googlenews concernant la France (filtre) :

```
publish feed france : for $i in googlenews where $i contains 'France' return $i
    as expires september 2009
```

- Flux avec toutes les nouvelles sur mes joueurs favoris (jointure) :

```
publish feed newsmyplayers : for $i in googlenews, $p in myplayers where $i
    contains $p/name return $i
```

- Publication d'un flux par joueur favori avec les news le concernant :

```
for $p in myplayers publish feed $p/name : for $i in googlenews where $i contains
$p return $i
```

ou (en utilisant le flux newsmyplayers) :

```
for $p in myplayers publish feed $p/name : for $i in newsmyplayers where $i
    contains $p return $i
```

- Séparation des flux par source (split) :

```
for $s in distinct googlenews/source publish feed $s/name : for $i in googlenews
    where $i/source = $s return $i
```

- Effacement d'une publication :

```
delete feed france
```

- Effacement d'un ensemble de publications :

```
for $s in distinct googlenews/source delete feed $s/name
```

3. Catalogue des utilisateurs

Le catalogue contient les informations suivantes sur les différentes utilisateurs d'un système :

Données utilisateurs

- login
- mot de passe
- groupes : subscriber, publisher, admin
- nom, prénom
- adresse email

Services

- enregistrement/modification d'un utilisateur
- dé/connexion d'un utilisateur
- effacement d'un enregistrement d'utilisateur

Langage

```
register user <login>                  : login unique
         password <password>           : mot de passe (optionnel)
         groups <groupes>              : groupes d'appartenance
         name <name>                   : nom (optionnel)
         firstname <firstname>         : prénom (optionnel)
         email <email>                 : adresse email
         sms <phone>                   : numéro de téléphone pour la
                                         réception de messages SMS (optionnel)
         roses <url>                   : adresse du pair ROSES optionnel)
         keywords <keywords>           : mots clés (optionnel)
```

```
login user <login>
         password <mot de passe>
```

```
logout
```

```
delete user <login>
         password <password>           : mot de passe (optionnel)
```

Exemples

```
register user toto password hello groups subscriber email toto@mail.fr keywords
   football
```

```
delete user toto password hello
```

4. Catalogue de souscriptions

Données

- login utilisateur
- nom de la publication
- format de diffusion : le format de diffusion définit le format de données et le protocole utilisé
- contraintes de fraîcheur / complétude (optionnel)

Services

- enregistrement de souscriptions
- effacement de souscriptions
- afficher toutes les souscriptions d'un utilisateur

Langage

```
subscribe to feed <name>              : nom de la publication
         format <format>              : format de diffusion
         refresh <fréquence>          : fréquence de rafraîchissement
                                        (optionnel)
```

- Effacement d'une souscription :

```
delete subscriptions to feed <name>   : nom de la publication
         with format <format>                  : optionnel (par défaut : tous)
```

- Désactivation d'une souscription :

```
disable subscriptions to feed <name> with format <format>
```

- Réactivation d'une souscription :

```
enable subscriptions to feed <name> with format <format>
```

Exemples

- Souscription au flux france au format RSS avec un rafraîchissement (poll) par jour :

```
subscribe to feed france format RSS refresh daily
```

- Souscription au flux france avec un mail par jour (send) :

```
subscribe to feed france format email refresh (at most 1 day and more than 5
```

```
    items)
```

- Souscription à tous les flux concernant les football :

```
for $f in publications() where $f/keyword = football subscribe to $f format SMS
    refresh (at most 1 day and more than 5 items)
```

- Effacement de toutes les souscriptions (d'un utilisateur) au flux france :

```
delete subscriptions to feed france
```

- Effacement de toutes les souscriptions (d'un utilisateur) au flux france au format RSS :

```
delete subscription france format RSS
```

- Effacement de toutes les souscriptions d'un utilisateur :

```
for $f in subscriptions() where $f/login = toto delete subscription $f
```

D Stockage

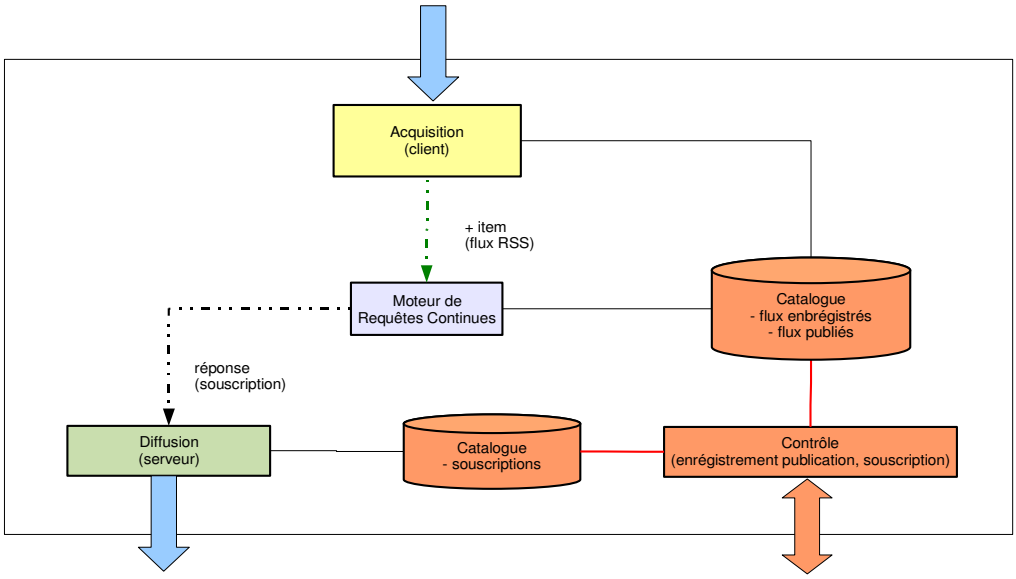
L'utilisation de la couche “stockage” dépend du modèle d'exécution des requêtes (et de l'application visée). On peut distinguer entre deux modèles d'exécution complémentaires :

1. Modèle “requêtes continues” : les publications correspondent à des requêtes continues sur des flux de données. Le besoin de stockage se limite à la gestion de fenêtres temporaires qui sont généralement limité en taille. Ce modèle semble bien adapté à des applications de type diffusion et surveillance avec des requêtes continues “simples”.
2. Modèle “requêtes temporelles + triggers” : On considère que les items d'entrée sont stockés dans un entrepôt de données et les publications correspondent à des requêtes temporelles avec des opérations de fenêtrage sur cet entrepôt. La dimension “flux” est implantée sous forme de triggers qui déclenchent ces requêtes. Ces requêtes peuvent combiner les données issues de l'entrepôt ROSES et des données statiques utilisées à travers des adaptateurs. Ce modèle semble mieux adapté à des applications d'archivage et d'enrichissement avec des requêtes complexes.

E Annexe

1. Modèles d'exécution

Modèle d'exécution « Requêtes continues »



Modèle d'exécution « Requêtes temporelles + entrepôt »