



ROSES : Un moteur de requêtes continues pour l'agrégation de flux RSS à large échelle

Jordi CREUS TOMÀS

Directeurs : Bernd AMANN et Dan VODISLAV

7 Décembre 2012

Laboratoire d'Informatique de Paris 6 (UPMC)



Plan

- Introduction
- Langage ROSES et algèbre
- Optimisation multi-requête de requêtes continues
- Évaluation expérimentale
- Optimisation dynamique
- Travaux connexes
- Conclusion et perspectives

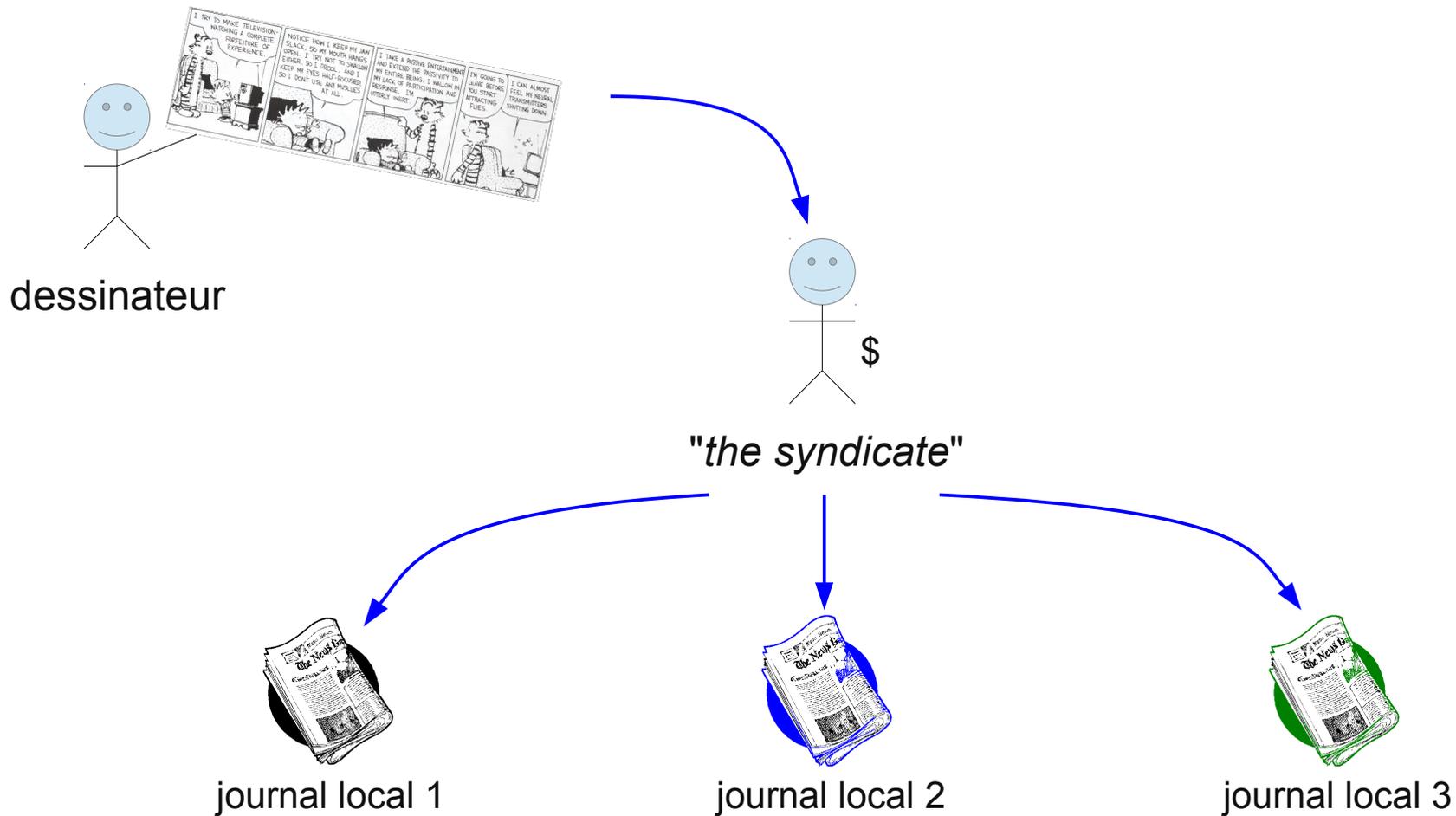
Introduction

Projet ROSES

- Really Open and Simple Web Syndication
- Projet ANR 07-MDCO-011
- *Objectif* : définition services de syndication pour la localisation, interrogation, génération, composition et personnalisation de flux RSS
- *Partenaires*
 - LIP6, CNAM, PRISM, LSIS et 2or3things
- <http://www-bd.lip6.fr/roses>

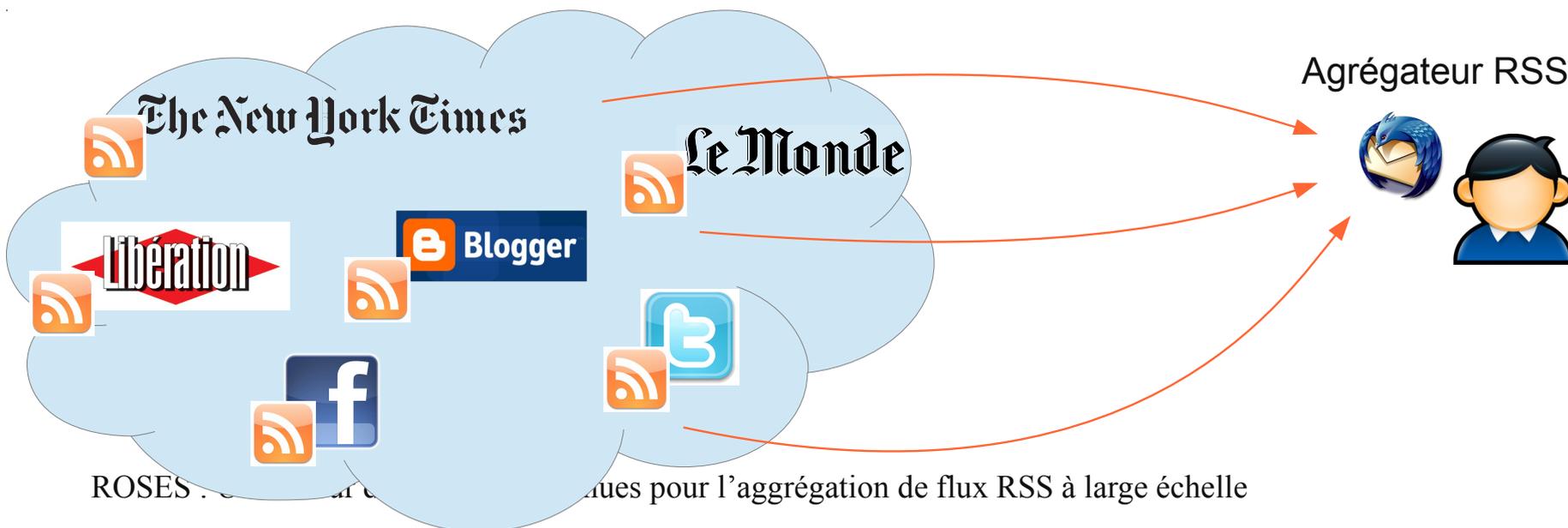
Histoire de la syndication

Début XX^{ème} siècle (USA)



Syndication web

- En 1999, Netscape publie RSS 0.9
- Really Simple Syndication
- *Syndication web*
 - Forme de syndication dans laquelle des contenus d'un site sont mis à disposition d'autres sites
 - Consiste à mettre à disposition des flux RSS avec l'en-tête du contenu récemment ajouté au site web (dernières news ou derniers post d'un blog)



Syndication web

- En 1999, Netscape publie RSS 0.9
- Really Simple Syndication
- *Syndication web*
 - Forme de syndication dans laquelle des contenus d'un site sont mis à disposition d'autres sites
 - Consiste à mettre à disposition des flux RSS avec l'en-tête du contenu récemment ajouté au site web (dernière news ou dernier post d'un blog)

<http://www.liberation.fr/feeds/politique.rss>



```
...  
<item>  
  <title>Copé et Fillon, castagnes de fin de campagne</title>  
  <link>http://www.liberation.fr/politiques/2012/11/11/cope-et-fillon-castagnes-de.html</link>  
  <description>UMP. Les candidats pour la tête du parti organisent chacun leur dernier  
  gros meeting cette semaine pour mobiliser avant le vote de dimanche.  
  Au terme d'un interminable marathon électoral... </description>  
  <pubDate>Sun, 11 Nov 2012 21:46:53 GMT</pubDate>  
  <dc:creator>Alain Auffray</dc:creator>  
</item>
```

Objectifs

- Agrégateur RSS
 - Des millions de flux → collections de flux
 - Beaucoup d'information → filtrage par contenu
- Agrégateurs actuels – problèmes
 - Opérations de filtrage basiques (mots clés) – sauf Yahoo! pipes
 - Pas d'agrégation par contenu
 - Stockage/archivage des données *protégées*
- Agrégateur ROSES
 - Création (et partage) de nouveaux flux personnalisés
 - Traitement continu (sans stockage)
 - Passage à l'échelle (nombre de requêtes et de sources)
 - Gestion dynamique des besoins et de l'information

Plan

- Introduction
- **Langage ROSES et algèbre**
- Optimisation multi-requête de requêtes continues
- Évaluation expérimentale
- Optimisation dynamique
- Travaux connexes
- Conclusion et perspectives

Modèle de données ROSES

- Données : items, flux, fenêtres
 - Item : titre, lien, description, datePub, *annotations*
 - Flux de couples (item, annotation)
 - Fenêtre d'items (jointure)
- Algèbre logique :
 - Flux/fenêtre = ensemble de triplets (estampille, item, annotation)
 - Inspirée de [Krämer,07] - algèbres temporelles
 - Opérateurs
 - Conservateurs : sélection, union, fenêtrage, jointure d'annotation
 - Altérants : transformation
 - Sémantique snapshot : règles de réécriture

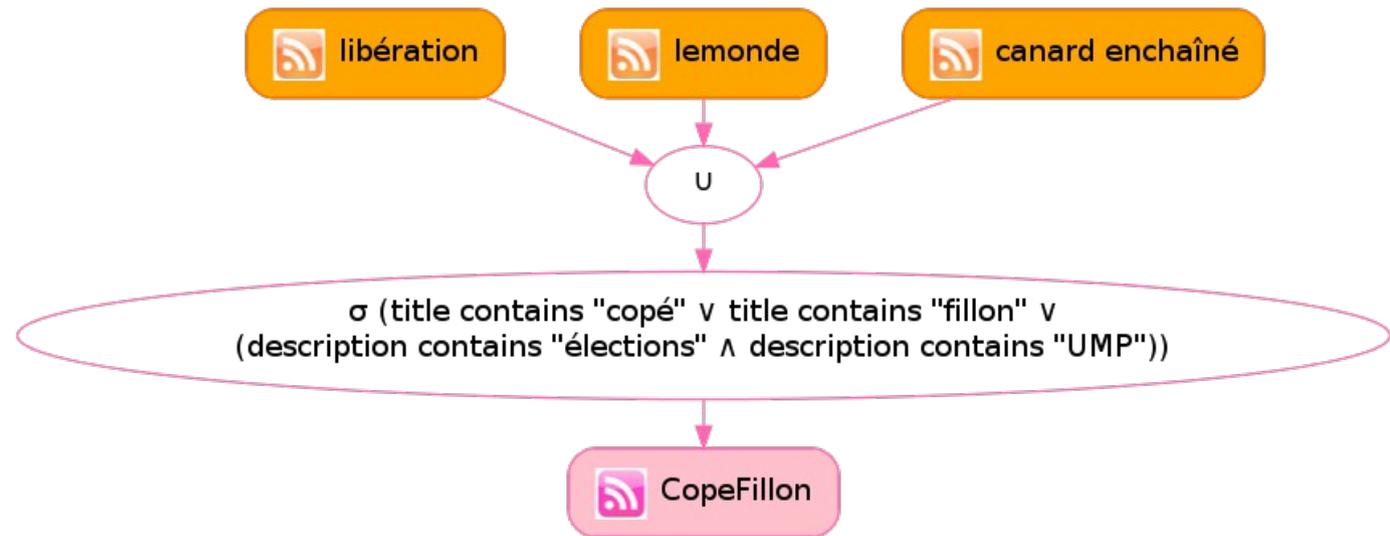
Langage ROSES

- Notation
 - Source : flux RSS ou Atom sur le web
 - Publication : flux produit par une requête ROSES sur un ensemble de sources et/ou d'autres publications
 - Souscription : souscription à une publication ROSES
- Langage ROSES
 - Langage déclaratif
 - Instructions : **register feed**, **create feed**, **subscribe to**
 - Opérateurs : sélection, union, fenêtrage, jointure (plus transformation)
 - Composition des publications

Langage de publication

- *Exemple 1* : filtrage

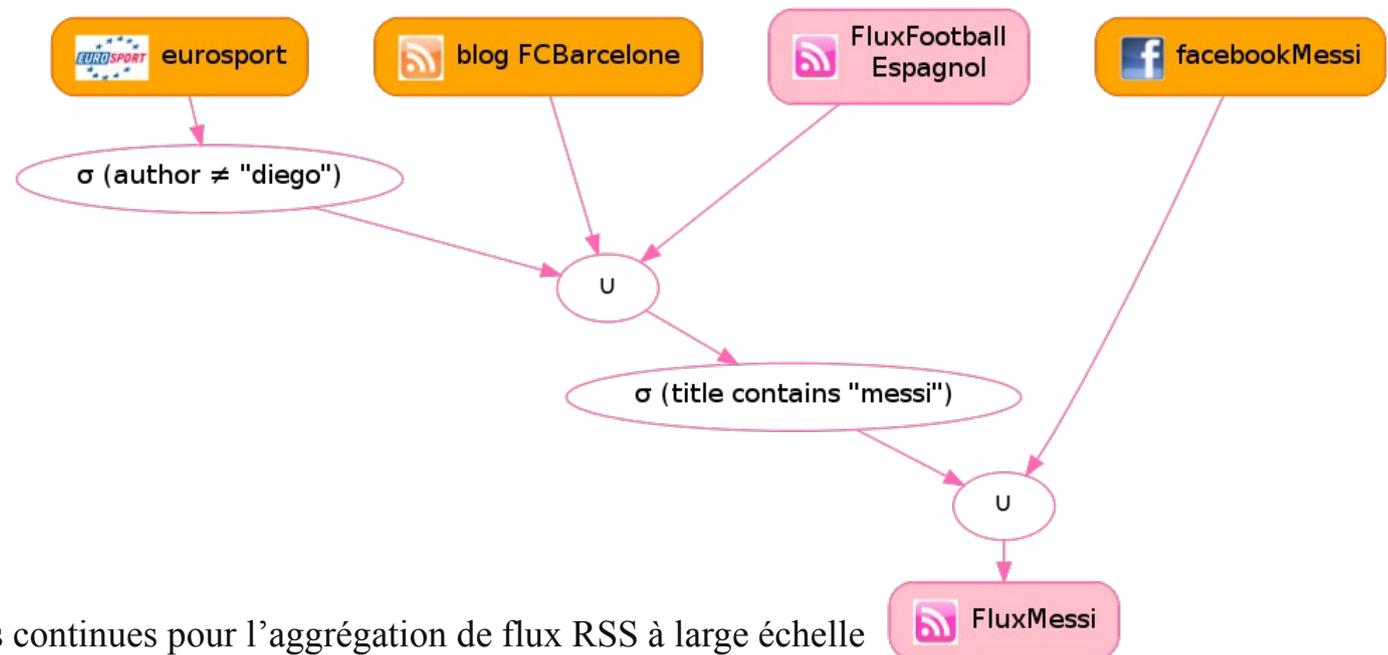
```
create feed CopeFillon
from liberation | lemonde | canardEnchaine
where title contains "copé" or title contains "fillon" or
(description contains "élections" and description contains "UMP");
```



Langage de publication

- *Exemple 2* : filtrage multiple et composition

```
create feed FluxMessi
from (eurosport as $e | blogFcbaselone | FluxFootballEsp) as $u | facebookMessi
where $e[author <> "diego"] and
    $u[title contains "messi"];
```

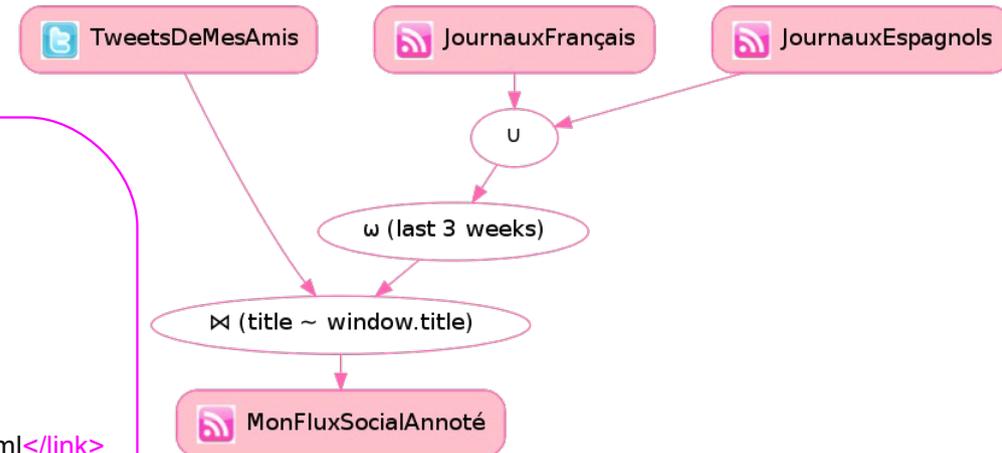


Langage de publication

- *Exemple 3* : jointure d'annotation

```
create feed MonFluxSocialAnnote
from TweetsDeMesAmis as $t
join last 3 weeks on (JournauxFrançais | JournauxEspagnols)
with $t[title similar window.title];
```

```
...
<item>
  <title>j'adore le feuilleton copé-fillon!</title>
  <link>http://twitter.com/stephaneski/statuses/223448426682138592</link>
  <description>j'adore le feuilleton copé-fillon!</description>
  <pubDate>Wed, 14 Nov 2012 9:51:12 GMT</pubDate>
  <roses:annotations>
    <item>
      <title>Copé et Fillon, castagnes de fin de campagne</title>
      <link>http://www.liberation.fr/politiques/2012/11/11/cope-et-fillon-castagnes-de.html</link>
      <description>UMP. Les candidats pour la tête du parti organisent chacun leur dernier
        gros meeting cette semaine pour mobiliser avant le vote de dimanche.
        Au terme d'un interminable marathon électoral...</description>
      <pubDate>Sun, 11 Nov 2012 21:46:53 GMT</pubDate>
    </item>
  </roses:annotations>
</item>
...
```

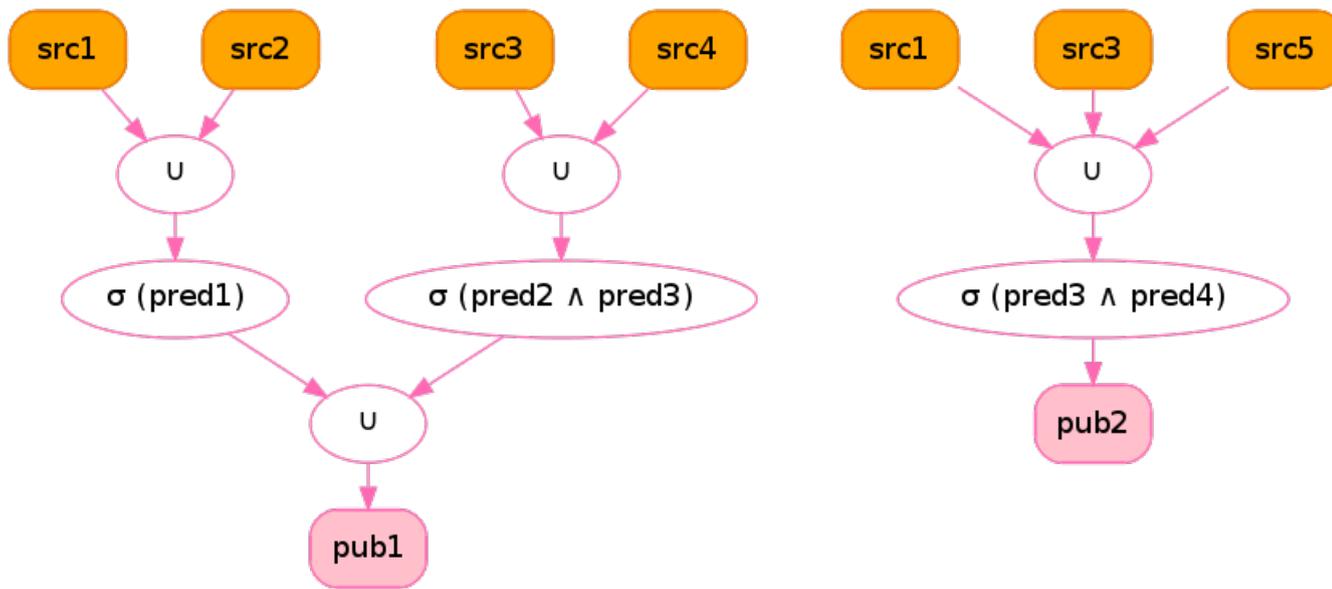


Plan

- Introduction
- Langage ROSES et algèbre
- **Optimisation multi-requête de requêtes continues**
- Évaluation expérimentale
- Optimisation dynamique
- Travaux connexes
- Conclusion et perspectives

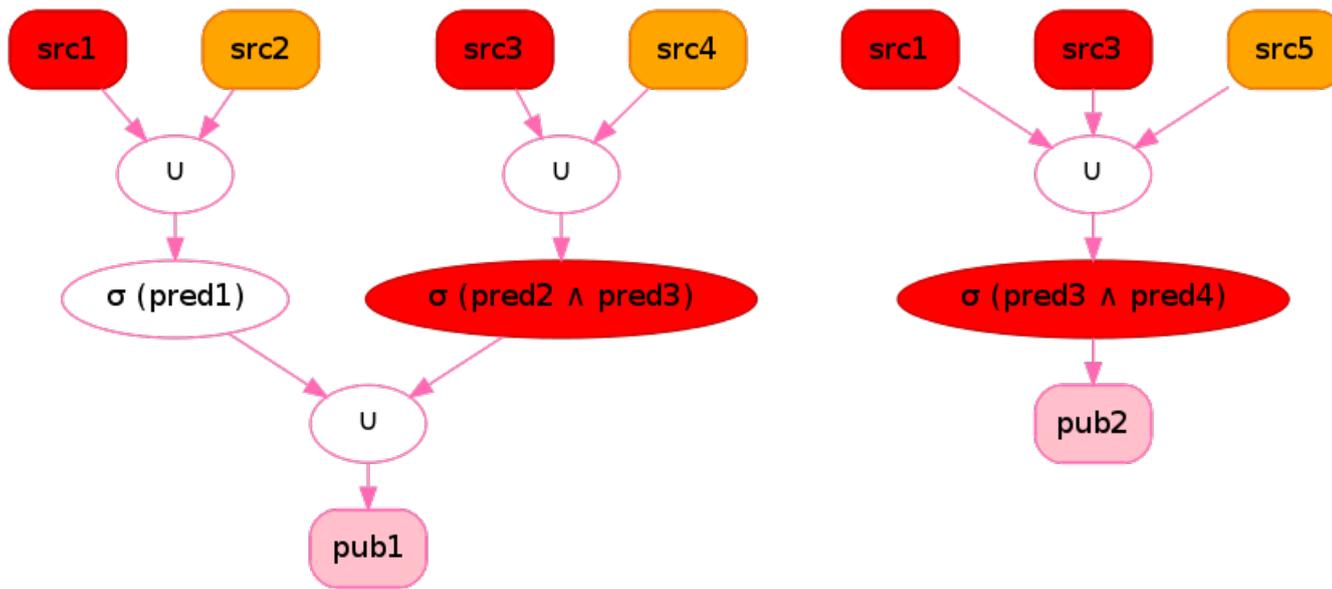
Optimisation multi-requête

- *But* : passage à l'échelle en termes de nombre de requêtes (publications) et des sources
- *Problème* : optimisation multi-requête
 - Optimisation des plans d'exécution par factorisation des opérateurs des requêtes "similaires"



Optimisation multi-requête

- *But* : passage à l'échelle en termes de nombre de requêtes (publications) et des sources
- *Problème* : optimisation multi-requête
 - Optimisation des plans d'exécution par factorisation des opérateurs des requêtes "similaires"



Problème : comment "matcher" des expressions similaires ?

Solution : forme normale

Optimisation statique

- Problème : optimisation *statique*
 - *Entrée* : ensemble de requêtes ROSES et modèle de coût d'opérateurs
 - *Sortie* : graphe multi-requête optimal

optimisation statique

Input: Q : ensemble de requêtes, M^C : modèle de coût

Output: G^* : graphe multi-requête optimal

// *Normalisation*

foreach $q_i \in Q$

normaliser q_i

générer un *graphe global normal* G^N en fusionnant toutes les requêtes

// *Factorisation*

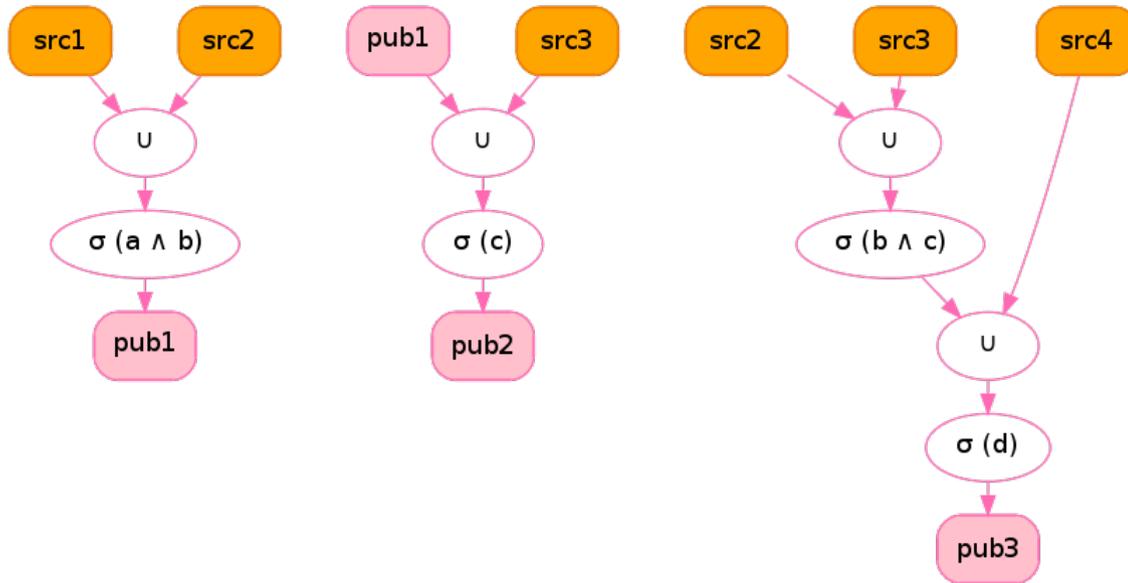
foreach source $src_i \in G^N$

générer un *graphe de subsomption de prédicats* G^S_i pour src_i

pondérer les arcs de G^S_i à partir du modèle de coût

trouver un arbre de filtrage de coût minimal sur G^S_i (arbre de *Steiner* minimal)

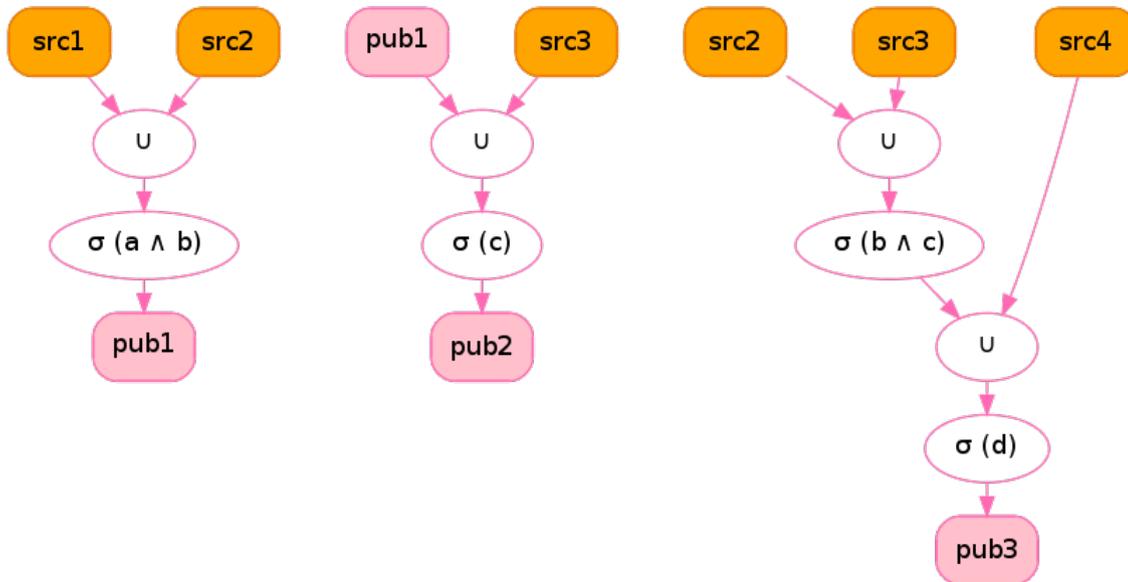
Normalisation



Normalisation

- ◆ Pousser sélections vers les sources
- ◆ Décomposition des publications
- ◆ Regrouper sélections en cascade dans un unique opérateur
- ◆ Transformer les prédicats de filtrage en CNF
- ◆ ...

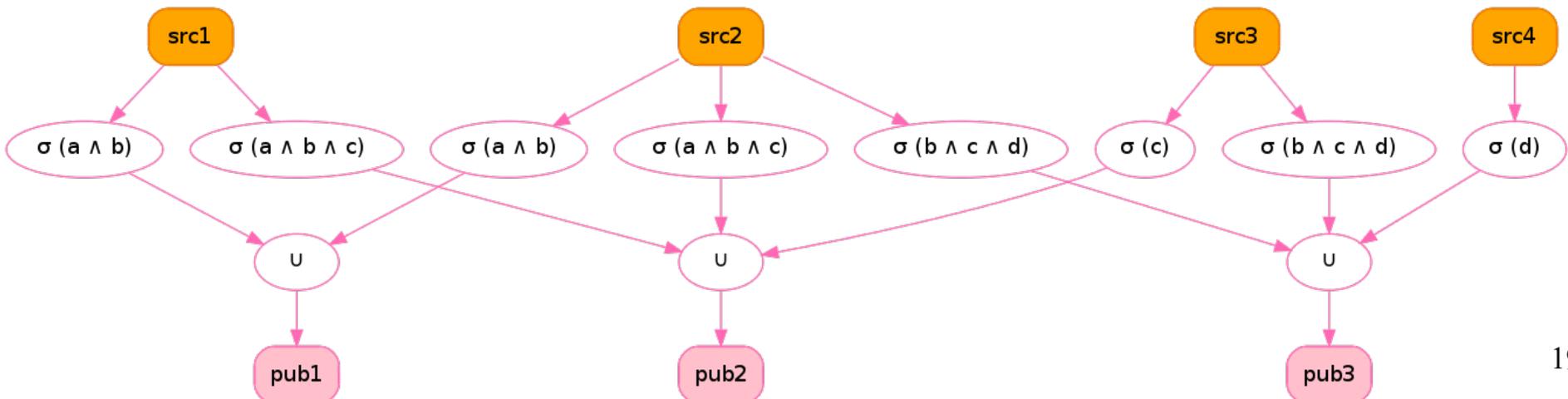
Normalisation



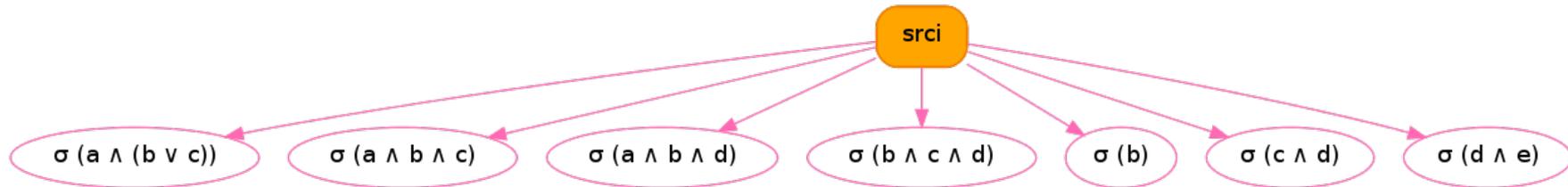
Normalisation

- ◆ Pousser sélections vers les sources
- ◆ Décomposition des publications
- ◆ Regrouper sélections en cascade dans un unique opérateur
- ◆ Transformer les prédicats de filtrage en CNF
- ◆ ...

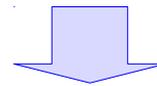
Grappe global normal



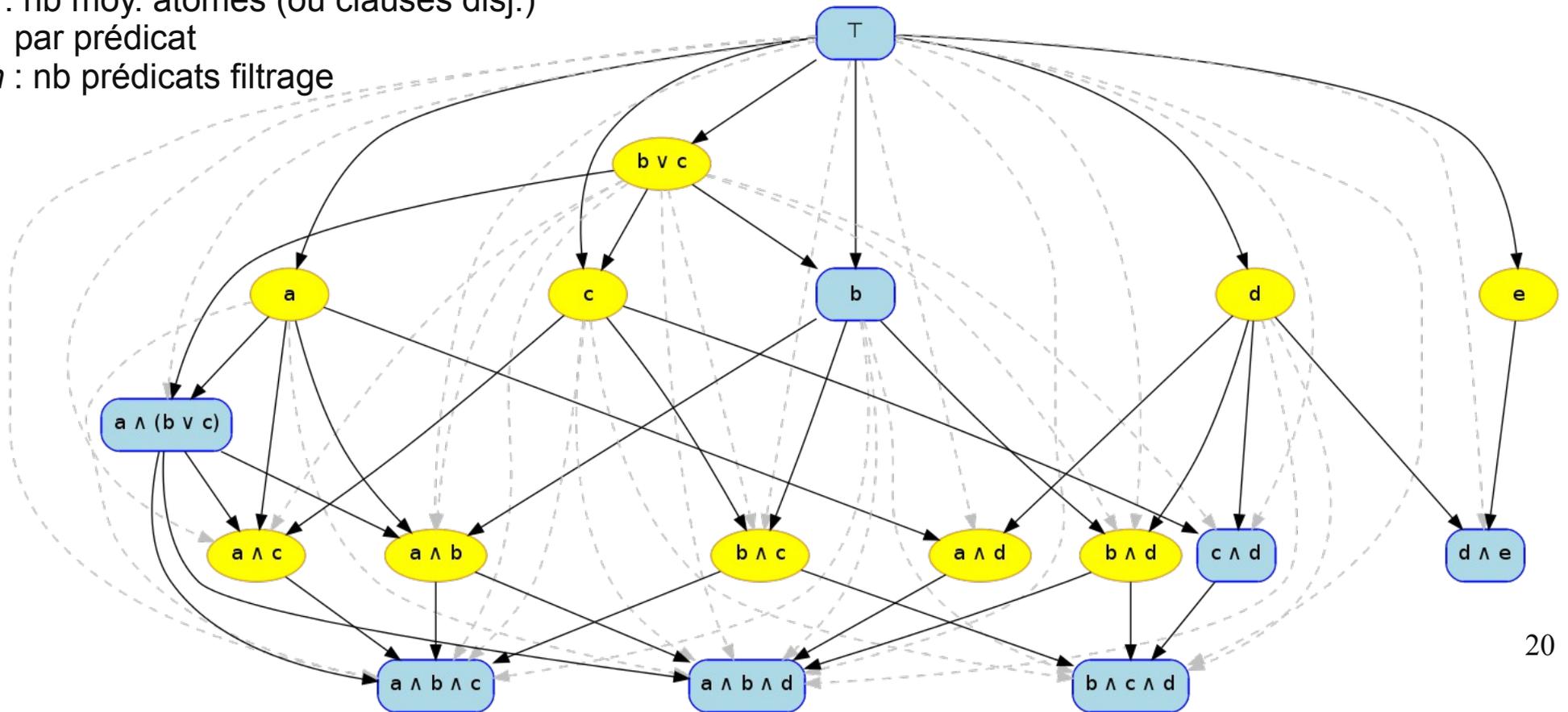
Graphe de subsomption de prédicats



taille $O(2^n \cdot m)$
 n : nb moy. atomes (ou clauses disj.)
 par prédicat
 m : nb prédicats filtrage



Génération graphe de subsomption



Graphe de subsomption de prédicats

$$\text{coût}(x \rightarrow y) = \text{sélectivité}(x, \text{src}_i) \cdot \text{débit}(\text{src}_i)$$

sélectivité(x, src_i) : fréquence du terme dans source src_i (dans passé récent)

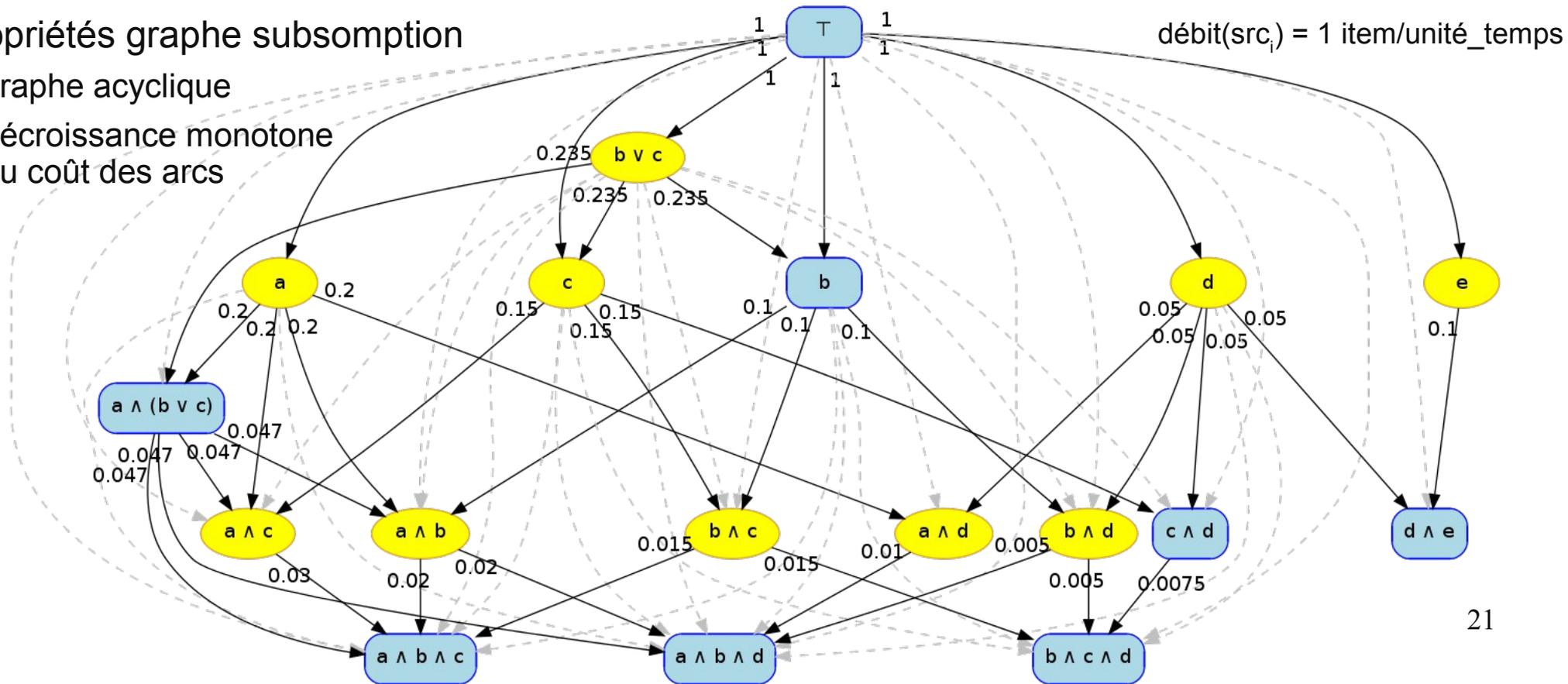
sélectivité($x \wedge y$) = sélectivité(x) · sélectivité(y)

sélectivité($x \vee y$) = sélectivité(x) + sélectivité(y) – sélectivité(x) · sélectivité(y)

...

Propriétés graphe subsomption

- graphe acyclique
- décroissance monotone du coût des arcs

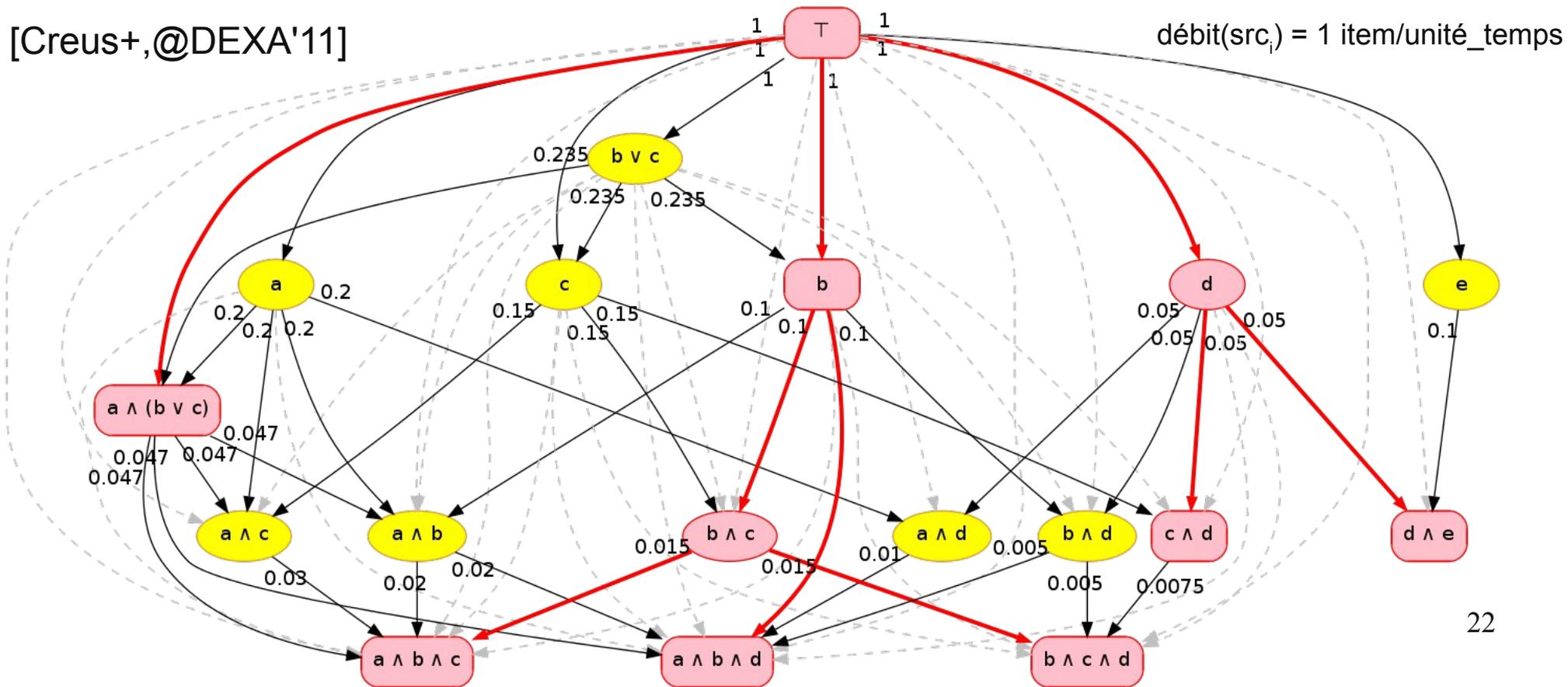


Arbre de *Steiner* minimal

Problème arbre de *Steiner* minimal

- Étant donné un graphe dirigé, étiqueté sur les arcs $G = (V, A, w^A)$, un nœud r , un ensemble T de nœuds *terminaux* ($T \subseteq V$)
- Trouver un arbre t^* de racine r , couvrant tous les nœuds dans T et de coût minimal

[Creus+, @DEXA'11]



Arbre de prédicats optimal

- Problème NP-complet (algorithmes d'approximation)
- Algorithmes de factorisation
 - STA
 - VCA - 2 versions
 - avec graphe de subsomption
 - avec VCB
- STA (*Steiner Tree Algorithm*)
 - Basé sur algorithme d'approximation de [Charikar+,@SODA'98]
 - Taux approximation : $i \cdot (i - 1) \cdot k^{1/i}$ k : nb nœuds terminaux
 - Parcours exhaustif du graphe en cherchant des sous-arbres de Steiner guidé par une heuristique de densité
 - densité* : nb de nœuds terminaux dans le sous-arbre courant / coût du sous-arbre
 - Très coûteux

Arbre de prédicats optimal

- VCA (*Very Clever Algorithm*)

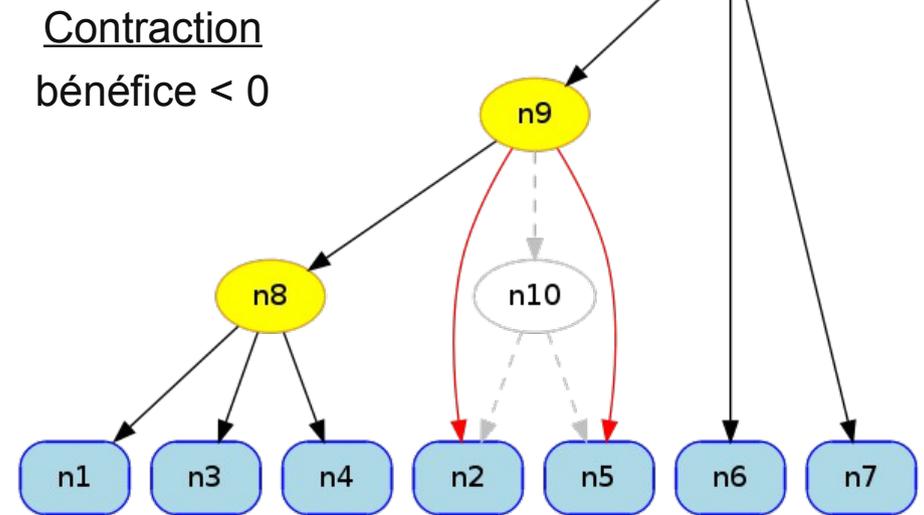
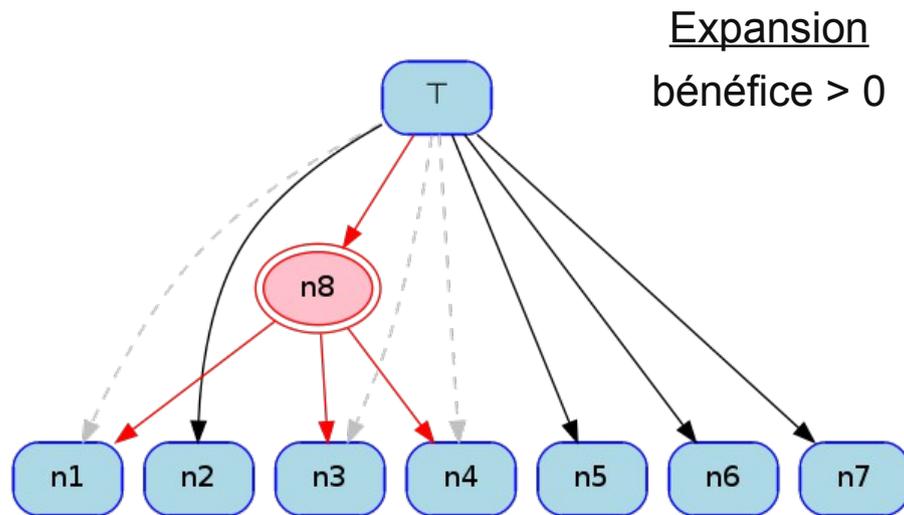
[Creus+,@BDA'11]

- Algorithme glouton *bottom-up*

- À chaque itération, 2 phases (expansion-contraction), guidé par heuristique de *bénéfice*

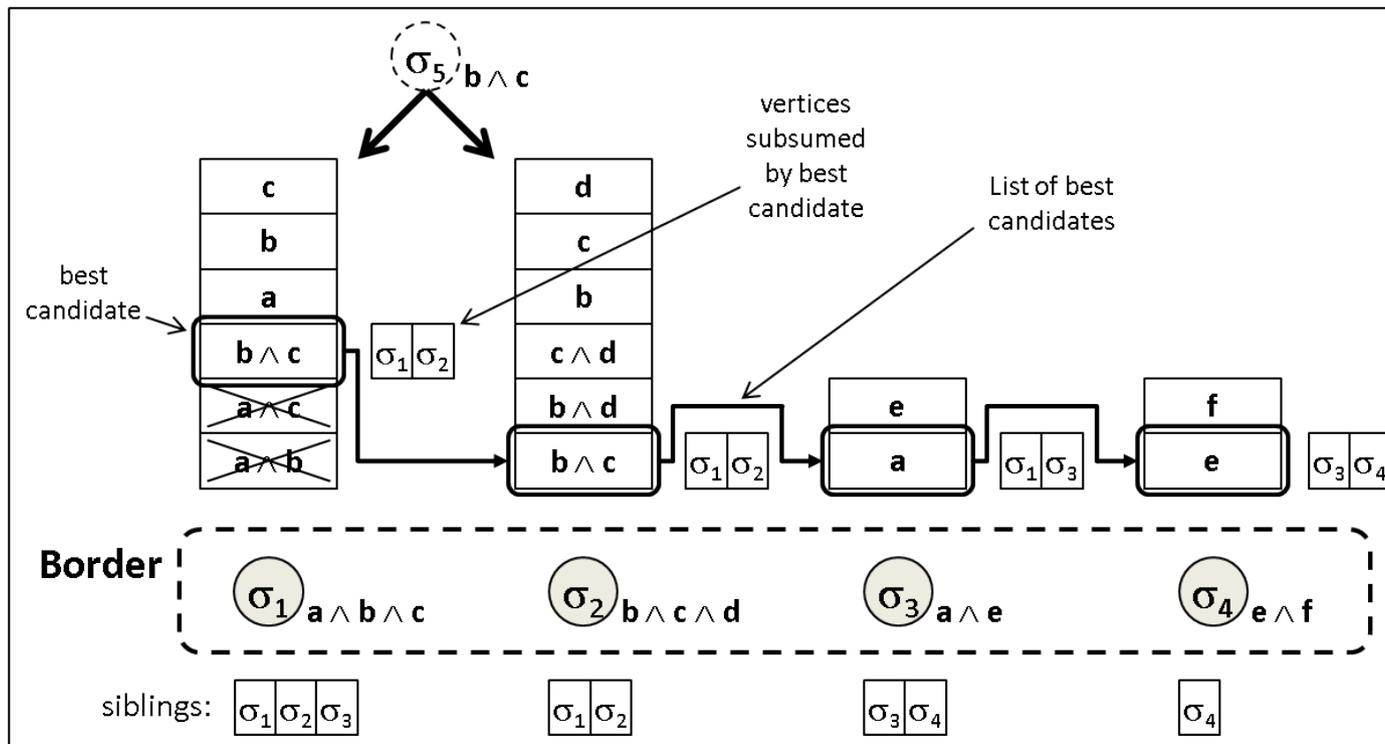
- $\text{bénéfice}(x, y) = (n - 1) \cdot \text{sélectivité}(y) - n \cdot \text{sélectivité}(x)$

- n : nb fils de y aussi subsumés par x



Arbre de prédicats optimal

- VCB (*Very Clever Border*)
 - Structure dynamique décroissante à la place du graphe complet
 - À chaque itération, nous générons *seulement les nœuds (arcs de subsomption) dont nous avons besoin* (triés par sélectivité)



Plan

- Introduction
- Langage ROSES et algèbre
- Optimisation multi-requête de requêtes continues
- **Évaluation expérimentale**
- Optimisation dynamique
- Travaux connexes
- Conclusion et perspectives

Évaluation expérimentale

- Nous avons crawlé pendant un mois 1300 flux de journaux et blogs anglais et français
 - ~ 190.000 items
 - Extraction de mots-clés : ~ 300 mots-clés/source
- Générateur de requêtes ROSES
 - Basé sur *Générateur souscriptions RSS* (Nelly Vouzoukidou)
 - Forme requêtes

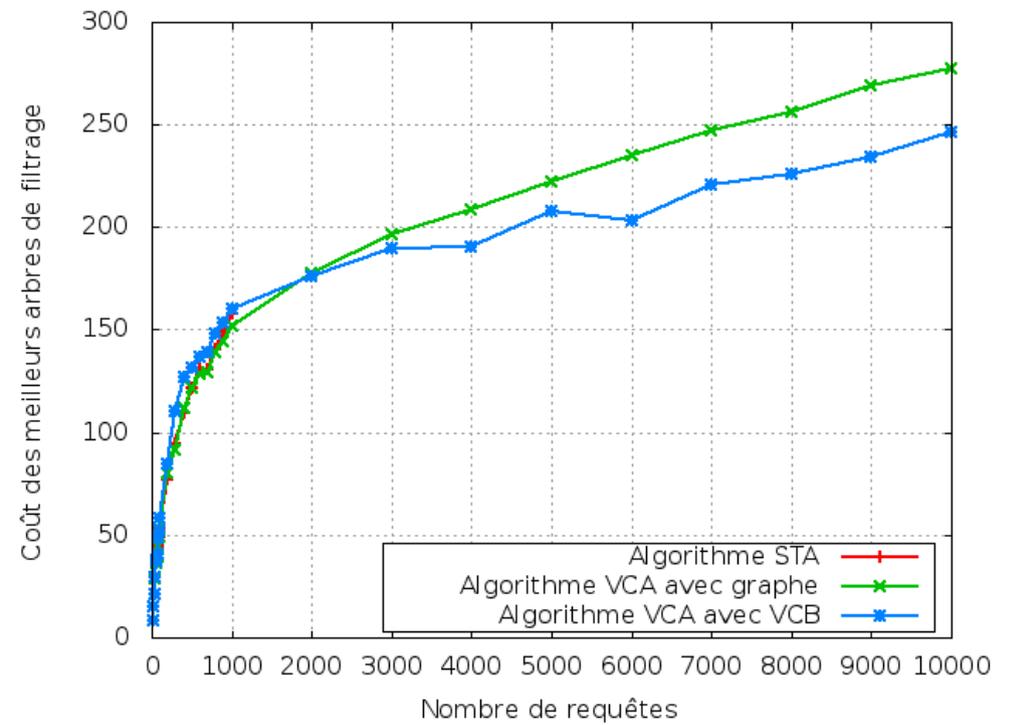
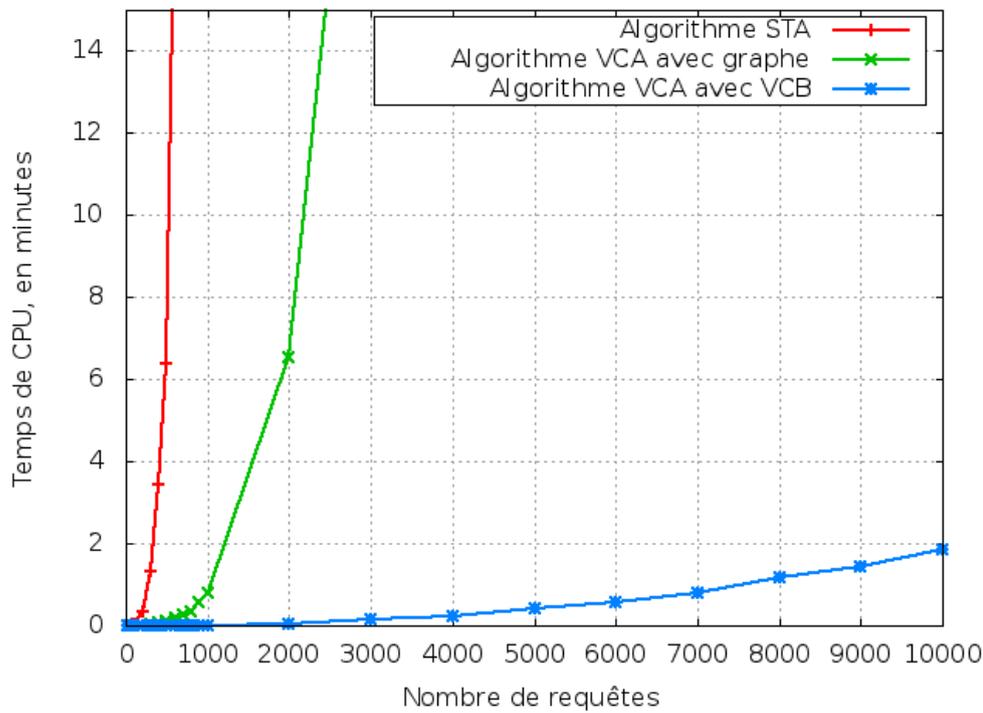
```
create feed Publication_de_synthese,  
from src1 | src2 | src3 | ... | srcn  
where prédicat_en_CNF;
```
 - Paramétrage
 - $[s_q^{\min}, s_q^{\max}]$: nb min et max de sources par requête
 - $[d^{\min}, d^{\max}]$: nb min et max de clauses disjonctives dans prédicat
 - $[a^{\min}, a^{\max}]$: nb min et max de prédicats atomiques dans disjonctions
 - ...

Évaluation expérimentale

- *Expérience 1* : requêtes à prédicats conjonctifs simples

Jusqu'à 10.000 requêtes

Paramètres : $s_q = 1$, $d = [1,3]$, $a = 1$

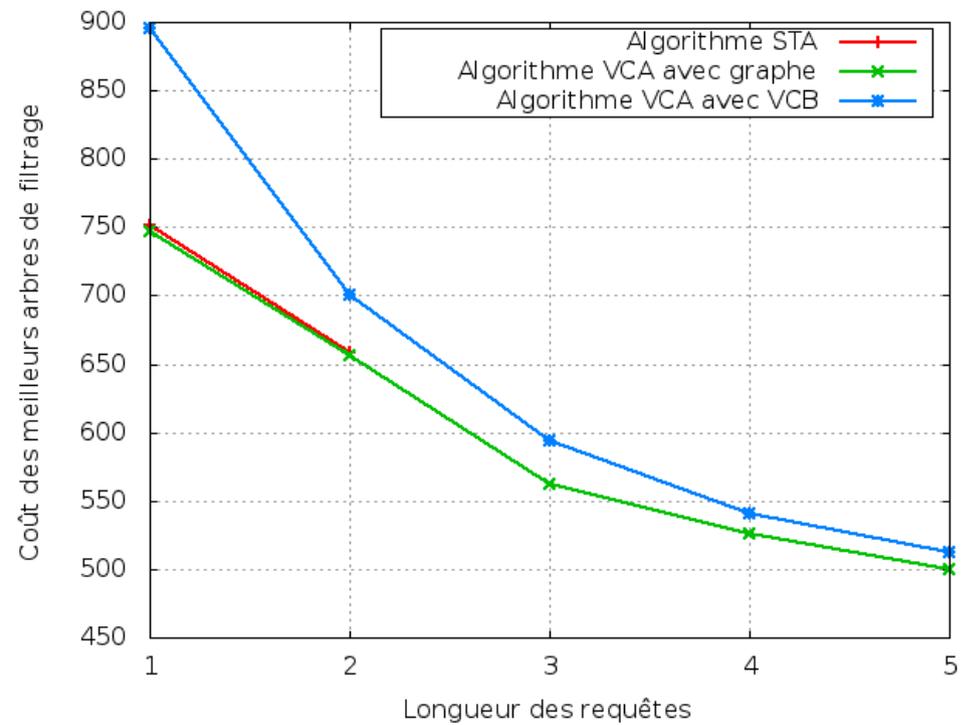
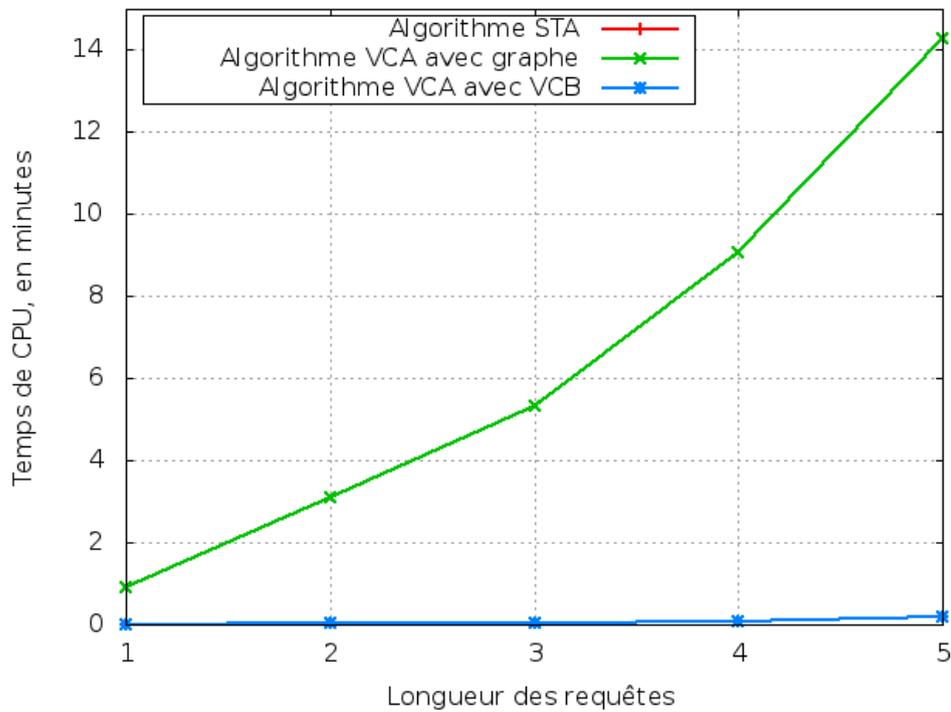


Évaluation expérimentale

- *Expérience II* : requêtes à prédicats complexes

1000 requêtes

Paramètres : $s_q = 1$, $d^{\min} = 1$, $d^{\max} \in [1, 5]$, $a^{\min} = 1$, $a^{\max} = 5$



Plan

- Introduction
- Langage ROSES et algèbre
- Optimisation multi-requête de requêtes continues
- Évaluation expérimentale
- **Optimisation dynamique**
- Travaux connexes
- Conclusion et perspectives

Optimisation dynamique

- Optimisation dans système gestion de requêtes continues : difficultés
 - Arrivée, départ et mise-à-jour des requêtes
 - Débit changeant des sources
 - Sélectivité changeant des termes de filtrage
- ⇒ Dégradation du graphe d'exécution
- Solutions :
 - réoptimiser à chaque changement → trop coûteux
 - réoptimiser après un seuil de dégradation
 - estimation périodique du degré de dégradation
 - estimation du seuil
 - optimisation « locale » :
 - Insertion des prédicats dans la « meilleure » position dans le graphe

Plan

- Introduction
- Langage ROSES et algèbre
- Optimisation multi-requête de requêtes continues
- Évaluation expérimentale
- Optimisation dynamique
- **Travaux connexes**
- Conclusion et perspectives

Travaux connexes

- *RUMOR: Rule-based Multi-query Optimization Framework* [Hong+,@EDBT'09]
 - Multi-opérateurs : regroupent les traitements des opérateurs similaires, produisent plusieurs sorties par opérateur
 - Traitement synchrone du sous-graphe regroupé
- *Optimization of continuous queries with shared expensive filters* [Widom+,@PODS'07]
 - Sur des filtres très chers (filtrage image, vidéo)
 - Ordonnancement des filtres
 - Basé sur arbre de décision
- *Near-optimal algorithms for shared filter evaluation in data stream systems* [Liu+,@SIGMOD'08]
 - Graphe biparti (requêtes – filtres atomiques)

Conclusion

- Contributions

- Langage d'interrogation orienté flux RSS (basé sur une algèbre formelle)
- Nouvelle technique d'optimisation multi-requête avec modèle de coût fondée sur la recherche d'un arbre de Steiner minimal
- Prototype fonctionnel système ROSES [Creus+,@CIKM'11]
<http://www-bd.lip6.fr/roses/doku.php?id=prototypes>

- Perspectives

- Validation expérimentale optimisation dynamique
- Ordonnancement (et factorisation) des jointures-fenêtres
- Traitement réparti graphe d'exécution des requêtes

Merci pour votre attention



Bibliographie

- [Creus+,@DEXA'11] Jordi Creus Tomàs, Bernd Amann, Nicolas Travers, and Dan Vodislav. **RoSeS: A Continuous Content-Based Query Engine for RSS Feeds**. In *Abdelkader Hameurlain, Stephen W. Liddle, Klaus-Dieter Schewe, and Xiaofang Zhou, editors, DEXA (2), volume 6861 of Lecture Notes in Computer Science, pages 203–218. Springer, 2011.*
- [Creus+,@BDA'11] Jordi Creus Tomàs, Bernd Amann, Vassilis Christophides, Nicolas Travers, and Dan Vodislav. **Optimizing large collections of continuous content-based RSS aggregation queries**. In *Bases de Données Avancées (BDA 2011), Rabat, Maroc, October 24th, 2011.*
- [Creus+,@CIKM'11] Jordi Creus Tomàs, Bernd Amann, Nicolas Travers, and Dan Vodislav. **RoSeS: a continuous query processor for large-scale RSS filtering and aggregation**. In *Craig Macdonald, Iadh Ounis, and Ian Ruthven, editors, CIKM, pages 2549–2552. ACM, 2011.*
- [Krämer,07] Jürgen Krämer. **Continuous queries over data stream - semantics and implementation**. *PhD thesis, 2007.*
- [Charikar+,@SODA'98] M. Charikar, C. Chekuri, T. Cheung, Z. Dai, A. Goel, S. Guha, and M. Li. **Approximation algorithms for directed Steiner problems**. In *Proceedings of the ninth annual ACM-SIAM symposium on Discrete algorithms, SODA '98, pages 192–200, Philadelphia, PA, USA, 1998. Society for Industrial and Applied Mathematics.*
- [Hong+,@EDBT'09] Mingsheng Hong, Mirek Riedewald, Christoph Koch, Johannes Gehrke, and Alan Demers. **Rule-based multi-query optimization**. In *Proceedings of the 12th International Conference on Extending Database Technology: Advances in Database Technology, EDBT '09, pages 120–131, New York, NY, USA, 2009. ACM.*
- [Widom+,@PODS'07] Kamesh Munagala, Utkarsh Srivastava, and Jennifer Widom. **Optimization of continuous queries with shared expensive filters**. *Proceedings of the twenty-sixth ACM SIGMOD-SIGACT-SIGART symposium on Principles of database systems - PODS '07, page 215, 2007.*
- [Liu+,@SIGMOD'08] Zhen Liu, Srinivasan Parthasarathy, Anand Ranganathan, and Hao Yang. **Near-optimal algorithms for shared filter evaluation in data stream systems**. In *Proceedings of the 2008 ACM SIGMOD international conference on Management of data - SIGMOD '08, page 133, New York, New York, USA, June 2008. ACM Press.*