Définition du cahier des charges !!!! Montrer que l'on a compris les Sigles, la démarche, le cahier des charges. Montrer que l'on a suivi la liste au tableau.

Problématique et argumentation pour/en deux options:

1) afficher une synthèse de différents flux rss distant et d'adresse connus

2) rendre accessible la synthèse à des clients distants

API fournit a tout les clients: plusieurs fonctionnalités :

* rendre une synthèse de deux flux dont l adresse avait été enregistré au préalable
* la même avec le nombre d'élément à retourner
* avec les flux à regrouper
* les dates

cette api tourne sur TOMCAT car elle utilise le java pour ces traitements

serveur a redimensionner selon le nombre de trafic si l api a du succès

Elle fournit en sortie un fichier xml ou d'autre type comme Json.

Mais comment ? :

Appel au flux rss externe par liens avec REST (passé en paramètres ou entré en dur) il récupère un fichier

Ce fichier est parsé à l aide de SAX on le traite pour instancier des classes java.

SAX: Il est préférable de l'utilisé par rapport à DOM car nous sommes intéressé qu'aux flux/éléments les plus récents. Nous n'avons en aucun cas besoin de lire le contenu. Le SAX est le moyen le plus rapide de savoir si une nouvelle new/flux a été publié. Une mise en cache des éléments précédemment parsé permettra un gain de temps sur ses accès extérieurs : tous les éléments ne seront pas parsés à chaque demande.

Ces différents objets sont regroupé dans une liste (java) selon des paramètres prédéfinis (tri par date titre...)

On utilise SAX ou Gson pour générer un xml ou json (desérialiser)

Si le client veux un historique des flux, (ex si les autres flux sont a durée de vie limité on peut souhaiter les conserver) on peut mettre en place une base de données dans laquelle tous les flux seront sauvegardés. L'appel à la base de données rendra l'api plus lente mais l'historique sera présent. La BD nous permettra d'implémenter une recherche de flux et rendre un service supplémentaire au client.

L'utilisation d'un serveur augmentera le coût final

2) 2 choix s'offrent a nous afin de répondre au cahier des charges sur l'affichage des flux par rendu HTML dans la balise div du client:

- se découpler au maximum de l'api 1 et se considérer comme un client de cette dernière

Ainsi on distinguera les 2 sorties possibles par 2 API différentes sachant que la 2 utilisera la 1 pour obtenir le fichier xml utilisé.

- regrouper les 2 fonctionnalités dans la même API (1)

Pour cela, on se cantonnera à une seule API afin de simplifier les accès et le nombre de serveurs.

L'URL de base commune devra se différencier par une fin d'URI différente afin de permettre l'accès à l'une ou l'autre des actions (sortie xml ou html).

Choix retenu : regrouper dans la même API.