

# Projet de Web Data Management

Olivier Marty & Shendan Jin  
22 février 2016

# Outline

1. But du projet
2. Détails techniques
3. Description des étapes
4. Ajouter de nouvelles sources

# But du projet

- Utiliser différentes sources d'informations sur internet
  - l'état de différents moyens de transport (vélo, métro, etc)
  - les événements à venir (Google calendar, Gmail)
- Prévoir les problèmes pour un déplacements
  - trafic ?                      → station vélo pleine ?
- Envoyer une notification en cas de problème
  - par SMS, email.

# Détails techniques

- Implémenté en Python3
- Web scrapping pour l'état des lignes RATP et Transilien
- Utilise l'API publique de jcdecaux\_vls (vélo libre service)
- Utilise l'API d'OpenStreetMap pour la localisation d'adresses
- Utilise les API de Google Calendar et Gmail pour la récupération d'événements

# I. Lister les événements à venir

- Lister les événements auquel l'utilisateur va se rendre
  - Une classe avec 3 attributs : (date, location, description)
- Sources : Google Calendar & Gmail
  - API de Google calendar et Gmail

# Analyse des emails

- Code actuel : *proof of concept*

→ Pour Gmail, seul les mails de la forme suivante sont reconnus

“Rendez-vous

le 22/02/2016 08h45

à Université Paris Diderot

pour le cours de WDM”

## II. Localiser l'adresse de l'événement

- L'API OpenStreetMap associe des coordonnées à l'adresse
- On recherche les stations les plus proches, pour chaque source de données
  - jcdecaux fournit la liste des stations avec les coordonnées
  - la RATP fournit des données statiques, desquelles on a extrait la liste des stations avec les lignes les traversant

# III. Lister les conditions de trafic

- des différentes sources de données
- pour les différentes entités (lignes, stations vélo)
- dans certains cas, un problème est soulevé
  - ligne interrompue (colis suspect...), travaux...
  - pour les vélos : la station est presque pleine ou vide

## IV. Envoyer une notification

- Chaque problème vient avec une description *human-readable*
  - “Ligne interrompue : malaise voyageur”
  - “Station okabé - le kremlin-bicêtre : plus que 2 places disponibles !”
- Le message est envoyé
  - par mail
  - par SMS (via l’API de free mobile)

# Ajouter de nouvelles sources de trafic

Chaque source doit fournir :

- une map *identifiant*  $\rightarrow$  *nom*
- une map *identifiant*  $\rightarrow$  *liste de positions*
- une fonction *identifiant*  $\rightarrow$  *Source*

Et c'est tout !

# Ajouter de nouvelles sources d'événements

Chaque source doit fournir une fonction qui génère une liste d'Event avec :

- id : identifiant unique pour l'événement
- date : date et heure
- location : adresse
- description

Place à la démo !