

# 1 Introduction

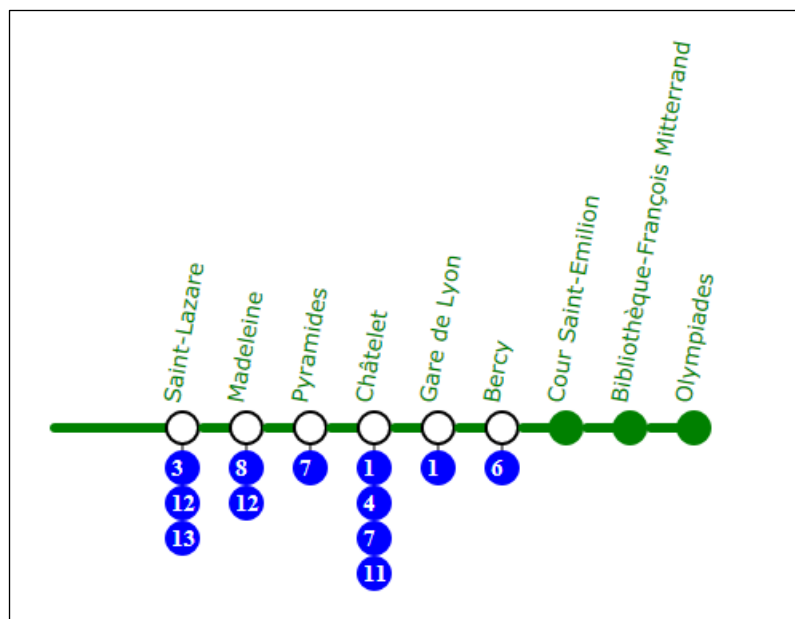
Le but de ce projet est de convertir par programme, au format **XML** (*Extensible Markup Language*), un ensemble de données brutes décrivant la structure du métro parisien, puis d'extraire de ces données converties, à l'aide de **XSLT** (*eXtensible Stylesheet Language*), des cartes des différentes lignes du métro – sur le modèle des cartes horizontales affichées dans les rames.

Le point de départ de ce projet est un fichier **CSV** (*Comma-separated values*) fourni avec l'énoncé du projet (*base\_ratp.csv*). Voici un exemple de la façon dont la ligne de métro numéro 14 est décrite dans ce fichier :

	A	B	C	D	E	F	G	H	I	J	K	L
1	station_id	station_name	station_desc	station_lat	station_lon	stop_sequence	route_id	service_id	direction_id	service_short	long_name_first	long_name_last
814	3064	Saint-Lazare	Rue de l'Arcade - 75108	48.8760678143526	2.32418810077101	1	1382497	2280413	0		14 SAINT-LAZARE	OLYMPIADES
815	2600	Madeleine	Rue De S <sup>an</sup> cze - 75108	48.8697947152178	2.32461201202227	2	1382497	2280413	0		14 SAINT-LAZARE	OLYMPIADES
816	2944	Pyramides	Rue de l'Echelle - 75101	48.8664969870386	2.3340942152591	3	1382497	2280413	0		14 SAINT-LAZARE	OLYMPIADES
817	2224	Châtelet	Th <sup>é</sup> âtre du Châtelet - 75101	48.8572036455044	2.34722471105584	4	1382497	2280413	0		14 SAINT-LAZARE	OLYMPIADES
818	2424	Gare de Lyon	Rue de Bercy - 75112	48.8446521509829	2.37310814754528	5	1382497	2280413	0		14 SAINT-LAZARE	OLYMPIADES
819	2084	Bercy	Place du Bataillon du Pacifi	48.8401922444035	2.37947725726606	6	1382497	2280413	0		14 SAINT-LAZARE	OLYMPIADES
820	2280	Cour Saint-Emilion	Cour Chamonard - 75112	48.8333137155784	2.38729970438351	7	1382497	2280413	0		14 SAINT-LAZARE	OLYMPIADES
821	2088	Biblioth <sup>é</sup> que-Fr <sup>an</sup> cois Mitterrand	au 62 rue du Chevalier	48.8298306952711	2.37612000934484	8	1382497	2280413	0		14 SAINT-LAZARE	OLYMPIADES
822	2740	Olympiades	Rue de Tolbiac - 75113	48.8269482819608	2.36703843338759	9	1382497	2280413	0		14 SAINT-LAZARE	OLYMPIADES
823	2740	Olympiades	Rue de Tolbiac - 75113	48.8269482819608	2.36703843338759	1	1382498	2280413	1		14 SAINT-LAZARE	OLYMPIADES
824	2088	Biblioth <sup>é</sup> que-Fr <sup>an</sup> cois Mitterrand	au 62 rue du Chevalier	48.8298306952711	2.37612000934484	2	1382498	2280413	1		14 SAINT-LAZARE	OLYMPIADES
825	2280	Cour Saint-Emilion	Cour Chamonard - 75112	48.8333137155784	2.38729970438351	3	1382498	2280413	1		14 SAINT-LAZARE	OLYMPIADES
826	2084	Bercy	Place du Bataillon du Pacifi	48.8401922444035	2.37947725726606	4	1382498	2280413	1		14 SAINT-LAZARE	OLYMPIADES
827	2424	Gare de Lyon	Rue de Bercy - 75112	48.8446521509829	2.37310814754528	5	1382498	2280413	1		14 SAINT-LAZARE	OLYMPIADES
828	2224	Châtelet	Th <sup>é</sup> âtre du Châtelet - 75101	48.8572036455044	2.34722471105584	6	1382498	2280413	1		14 SAINT-LAZARE	OLYMPIADES
829	2944	Pyramides	Rue de l'Echelle - 75101	48.8664969870386	2.3340942152591	7	1382498	2280413	1		14 SAINT-LAZARE	OLYMPIADES
830	2600	Madeleine	Rue De S <sup>an</sup> cze - 75108	48.8697947152178	2.32461201202227	8	1382498	2280413	1		14 SAINT-LAZARE	OLYMPIADES
831	3064	Saint-Lazare	Rue de l'Arcade - 75108	48.8760678143526	2.32418810077101	9	1382498	2280413	1		14 SAINT-LAZARE	OLYMPIADES

*base\_ratp.csv*

Dans ce cas, le but est qu'à l'étape finale, nous obtenions le plan suivant :



*plan.svg*

Pour arriver à ce résultat, une description graphique du plan en format **SVG** (*Scalable Vector Graphics*), nécessite l'exécution de plusieurs étapes. Ces étapes seront détaillées dans ce document.

## 2 Conversion des données CSV en XML

Dans la première étape, une décision sur une structure appropriée pour le fichier xml est requise. A ce stade, il est très important de s'assurer qu'une donnée n'est pas enregistrée plusieurs fois (par exemple, dans le fichier CSV, pour chaque station de chaque ligne, tous les détails de la station sont répertoriés : nom, adresse, etc., même si ces détails apparaissent déjà dans le fichier). De plus, il faut éviter les informations inutiles, informations qui peuvent être déduites d'autres informations du fichier xml.

Dans un premier temps, voici un exemple de représentation en xml pour une station de métro sur un parcours d'une ligne, par exemple, la station de métro Château de Vincennes qui est une des extrémités de la ligne 1.

Voici comment la station apparaît dans le fichier csv :

	A	B	C	D	E	F	G	H	I	J	K	L
	station_id	station_name	station_desc	station_lat	station_lon	stop_sequence	route_id	service_id	direction_id	service_short	long_name_first	long_name_last
2	2212	Château de Vincennes	Gare routière - 94080	48.8443251443686	2.44055234621182		1	1577980	2280236	0	1	CHATEAU DE VINC LA DEFENSE
51	2212	Château de Vincennes	Gare routière - 94080	48.8443251443686	2.44055234621182		25	1577981	2280236	1	1	CHATEAU DE VINC LA DEFENSE

*base\_ratp.csv*

Et voici à quoi ressemble la représentation dans le fichier xml dans le cas où il n'y a que cette station sur la ligne 1 et que la ligne 1 est la seule ligne du métro parisien :

```
<?xml version="1.0" encoding="UTF-8"?>
<ratp>
  <stations>
    <!--Liste de toutes les stations de métro-->
    <station id="2212">
      <nom>Château de Vincennes</nom>
      <adresse>Gare routière - 94080</adresse>
      <latitude>48.8443251443686</latitude>
      <longitude>2.44055234621182</longitude>
    </station>
  </stations>
  <services>
    <!-- Liste de tous les services offerts par le réseau de métro (liste de toutes les lignes) -->
    <service id="2280236">
      <numero>1</numero>
      <trajet id="1577980" direction="0">
        <station ref-id="2212" />
      </trajet>
      <trajet id="1577981" direction="1" />
    </service>
  </services>
</ratp>
```

Vous pouvez prêter attention aux détails suivants :

1. L'élément racine est **ratp**
2. Cet élément comporte deux fils : **gares**, **services** (chaque service est en fait une ligne du métro).
3. Les fils de l'élément stations, **station**, représentent chacun une station de métro. Le but est que les informations sur chaque station soient stockées ici une fois pour toutes, sans répéter toutes les informations pour chaque ligne chaque fois que cette station apparaît (comme cela se produit dans le fichier csv), mais une fois que les informations sont enregistrées ici, nous utiliserons plus tard l'**attribut id** pour effectuer une référence.
4. Chaque **service** a au moins 2 **trajets**, un dans chaque direction. Si l'itinéraire dans les deux sens est le même, la liste des stations n'est enregistrée qu'une seule fois, s'il y a des changements dans l'un des sens, seules les différences entre les itinéraires apparaîtront, et non l'intégralité de l'itinéraire une fois de plus.
5. Les stations terminales de la ligne qui apparaissent dans le fichier csv ont été omises car il s'agit d'une information qui peut être déduite.

Après avoir décidé de la structure du fichier xml, un code **Python** a été écrit pour effectuer la conversion des données du fichier csv vers un fichier xml. Le nom du fichier Python de notre projet : *extracteur.py*. Le code a été écrit en utilisant **xml.dom** : <https://docs.python.org/3/library/xml.dom.html> .

## 3 Schéma XML (fichier xsd)

Le fichier **xsd** est conçu pour vérifier que la structure du fichier xml est correcte et conforme au format spécifié. A cette occasion, nous avons créé, bien qu'il n'y avait aucune obligation de le faire dans le cadre du projet, un fichier **DTD** ayant le même objectif. Cependant, l'utilisation du fichier xsd est bien sûr à privilégier car il permet un meilleur contrôle du contenu du fichier xml.

## 4 Conversion des données XML en SVG

Comme décrit précédemment dans ce document, le but est un affichage graphique en SVG d'un itinéraire de ligne de métro. A cet effet, dans une première étape il a été décidé de l'affichage graphique à atteindre dans l'étape finale, notamment en écrivant un extrait de code qui permet de créer cet affichage graphique :

```

<?xml version="1.0" encoding="UTF-8" standalone="no"?>
<svg xmlns="http://www.w3.org/2000/svg"
    height="450" width="940" version="1.0"
    xmlns:xlink="http://www.w3.org/1999/xlink">

<line x1="20" y1="200" x2="600" y2="200" stroke="green" stroke-width="6" stroke-linecap="round"/>
<line x1="20" y1="200" x2="20" y2="269" stroke="black" stroke-width="1" stroke-linecap="round"/>

<circle cx="20" cy="225" r="10" fill="blue" stroke="blue" stroke-width="2"/>
<text transform="translate (20, 185) rotate(-45)" font-family="Verdana" font-size="14"
fill="green"> Nation </text>
<text x="17" y="230" font-family="serif" font-weight="bold" font-size="15" fill="white">1</text>

<circle cx="20" cy="247" r="10" fill="blue" stroke="blue" stroke-width="2"/>
<text x="17" y="252" font-family="serif" font-weight="bold" font-size="15" fill="white">6</text>

<circle cx="20" cy="269" r="10" fill="blue" stroke="blue" stroke-width="2"/>
<text x="17" y="274" font-family="serif" font-weight="bold" font-size="15" fill="white">9</text>

<circle cx="20" cy="200" r="10" fill="white" stroke="black" stroke-width="2"/>
<text transform="translate (20, 185) rotate(-45)" font-family="Verdana" font-size="14"
fill="green"> Nation </text>

<circle cx="60" cy="200" r="10" fill="green" stroke="green" stroke-width="2"/>
<text transform="translate (60, 185) rotate(-45)" font-family="Verdana" font-size="14"
fill="green"> Avron </text>

<circle cx="100" cy="200" r="10" fill="green" stroke="green" stroke-width="2"/>
<text transform="translate (100, 185) rotate(-45)" font-family="Verdana" font-size="14"
fill="green"> Alexandre-Dumas </text>

<circle cx="140" cy="200" r="10" fill="green" stroke="green" stroke-width="2"/>
<text transform="translate (140, 185) rotate(-45)" font-family="Verdana" font-size="14"
fill="green"> Philippe Auguste </text>

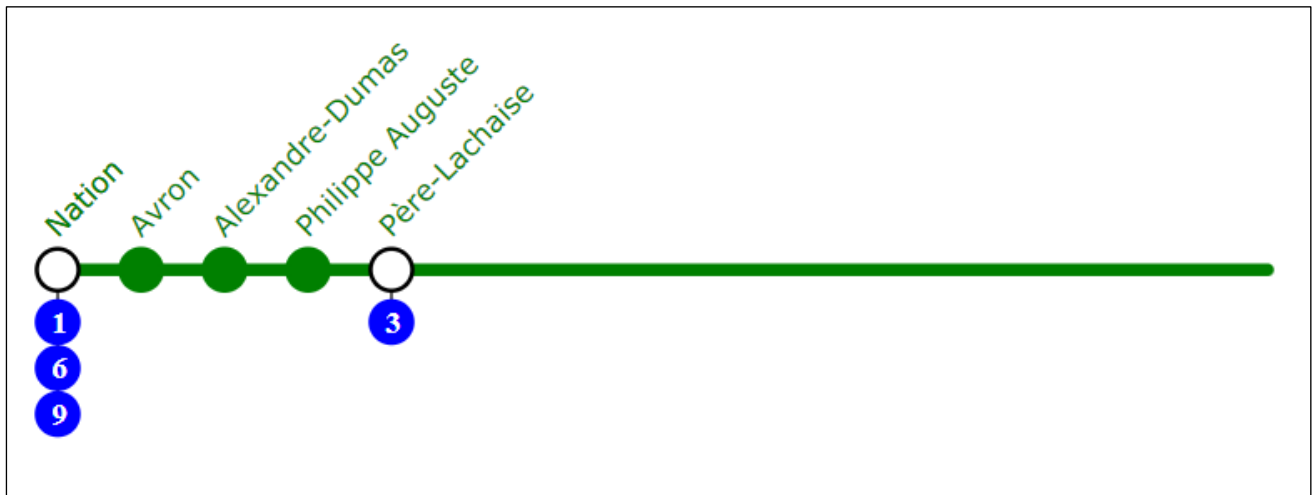
<line x1="180" y1="200" x2="180" y2="230" stroke="black" stroke-width="1" stroke-linecap="round"/>
<circle cx="180" cy="200" r="10" fill="white" stroke="black" stroke-width="2"/>
<text transform="translate (180, 185) rotate(-45)" font-family="Verdana" font-size="14"
fill="green"> Père-Lachaise </text>

<circle cx="180" cy="225" r="10" fill="blue" stroke="blue" stroke-width="2"/>
<text x="177" y="230" font-family="serif" font-weight="bold" font-size="15" fill="white">3</text>

</svg>

```

Ce code donne le résultat suivant :



Maintenant qu'il était clair quel était le résultat souhaité, il était possible de commencer la partie principale du projet : écrire du code XSLT qui convertirait le fichier XML en un affichage graphique en SVG du parcours de l'une ligne de métro. Le résultat apparaît dans le fichier *stylesheet.xsl*/accompagné de nombreux commentaires.