

Les bases de données avancées

2

- **Unité d'enseignement**

Les bases de données avancées 2

- **Année Académique**

2015/2016

- **Parcours**

INFOTEL – Informatique de Gestion – Licence 3

- **Enseignant**

Douwé Hallam Vincent (douwevincent@yahoo.fr)

Informations générales

- ORGANISATION DES COURS
Cours Magistral: 10h
Travaux Dirigés : 10h
Travaux Pratiques: 10h
- ORGANISATION DES EVALUATIONS
Un Contrôle Continu de 2h
Un TPE d'une semaine
Un Examen Final de 3h
Un Examen de Rattrapage de 3h

Objectifs du cours

- Introduction aux modèles de données semi-structurées
- Présenter le langage XML
- Présenter les langages du XML
- Présenter des certaines bibliothèques et outils XML

Plan du cours

- Introduction
- Présentation du XML
- Validation des documents XML
- Interrogation les documents XML
- Transformation des documents XML
- Intégration du XML dans les langages de programmation
- Le modèle JSON.

Bibliographie

- Hector Garcia Molina, Jeffrey D. Ullman, Jennifer Widom, Database systems the complete book, 2nd Edition, Prentice Hall, 2008
- Thomas Connolly, Carolyn Begg, Database Systems : A practical approach to design, implementation and management, Fourth Edition, Addison-Wesley, 2005
- Alexandre Brillant, XML Cours et Exercices, Eyrolles,
- Cours XML, ???

INTRODUCTION

Introduction

Dans les cours antérieurs, nous avons étudié les bases de données relationnelles.

Ces bases de données se fondent sur le modèle relationnel.

Dans ces types de bases de données, l'on a deux types de données :

- L'intention de la donnée. Ici l'on définit la structure et les contraintes à respecter : le schéma
- L'extension de la donnée. Ici ce sont les données proprement parlé.

L'intention évolue peu alors que l'extension change beaucoup

Introduction

Ainsi, avec les bases de données relationnelles l'on doit d'abord définir le schéma des données avant que l'on ne puisses y stocker des valeurs.

Les valeurs stockées dans cette base doivent satisfaire les contraintes imposées par le schéma.

Cette rigidité imposée par le schéma est ce qui a fait le succès du modèle relationnel. Il existe des implémentations très efficaces de ce modèle du principalement au fait que le schéma fournit des informations permettant de répondre rapidement à certains types de requêtes.

Introduction

L'avènement du WWW a conduit à de nouveaux types de manipulation des données.

Les données échangées sur Internet ont les caractéristiques suivantes :

- Les données sont de tout type.
- La structure des données est très hétérogène.
- Il y a très peu de contraintes sur les données manipulées : les données sont structurellement auto-descriptives et ne nécessitent pas une description indépendante de leur structure pour être interprétées

Introduction

Pour prendre en considération ces contraintes, l'on doit travailler avec les données semi-structurées.

Une donnée semi-structurée est une donnée qui peut être irrégulière ou incomplète et dont la structure peut changer rapidement et de manière non prédictible.

Les modèles semi-structurés jouent deux rôles prépondérantes :

- Intégration des données. Ils permettent de décrire les données similaires dans deux bases de données qui n'ont pas le même schéma.
- Échange de données. Ils servent de modèle pour les langages comme XML pour l'échange des données sur Internet.

Motivations

Avec les modèles semi-structurés, il n'y a pas de schéma distinct de la donnée. La donnée définit elle même son schéma.

L'intérêt pour les modèles semi-structurés est motivé par cette flexibilité. La donnée porte elle même le schéma et ainsi le schéma va évoluer automatiquement avec l'évolution de la donnée.

Intégration des données

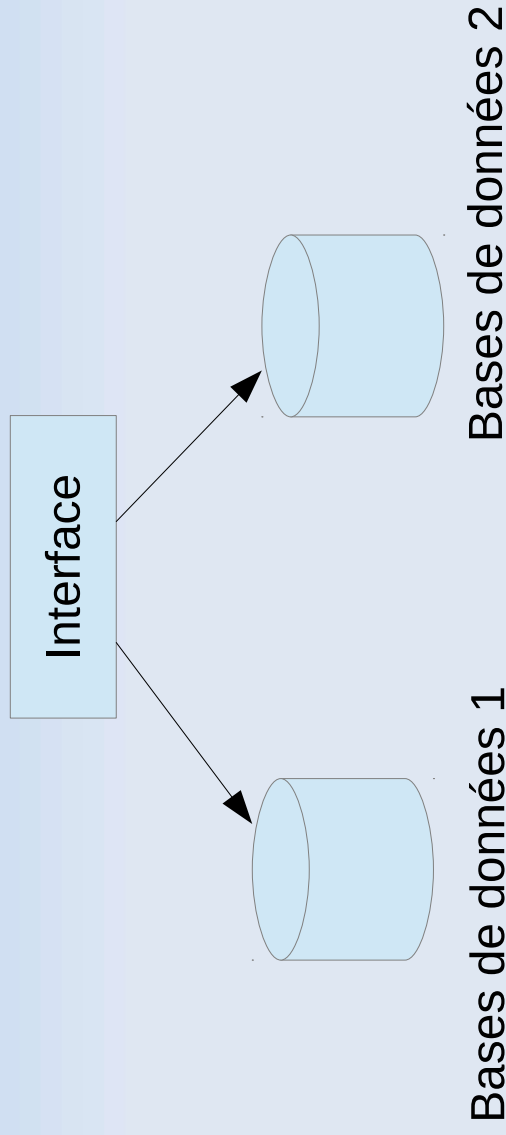
La prolifération des bases de données a créé un besoin d'intégration des données. L'idée est que si une données est disponible dans deux ou plusieurs bases de données l'on doit accéder à cette donnée comme si elle était disponible dans une seule base de données.

Si l'on dispose de bases de données qui ont le même schéma, alors il est très facile d'intégrer les données (en procédant à l'union des données).

La réalité est que rarement deux bases de données ont le même schéma.

Intégration des données

Une idée simple serait de définir une interface que l'on va implémenter pour accéder aux données d'une base de données.



La flexibilité et la nature auto descriptive des bases de données semi-structurées les rend adapté pour l'intégration des données.

L'échange de données

Les modèles semi-structurés sont bien adaptés pour échanges de données entre deux applications.

Souvent l'on se retrouve avec deux applications qui ont été conçu indépendamment de l'autre.

Comme il n'y a pas d'interface commun, l'on extrait les données de la première application que l'on va mettre en forme pour que ces dernières soient consommées par la deuxième application.

Représentation

Les données semi-structurées sont représentées sous forme d'ensemble de nœuds interconnectés. Un nœud est soit une feuille soit un nœud intérieur.

Les feuilles contiennent des données qui peuvent être de n'importe quel type atomique.

Les nœuds intérieurs ont un ou plusieurs arcs sortants. Les arcs ont des labels qui indiquent comment la source le l'arc relie à la destination de l'arc.

Il existe un nœud intérieur, qui n'a pas d'arc entrant, appelé racine représente la base de données.

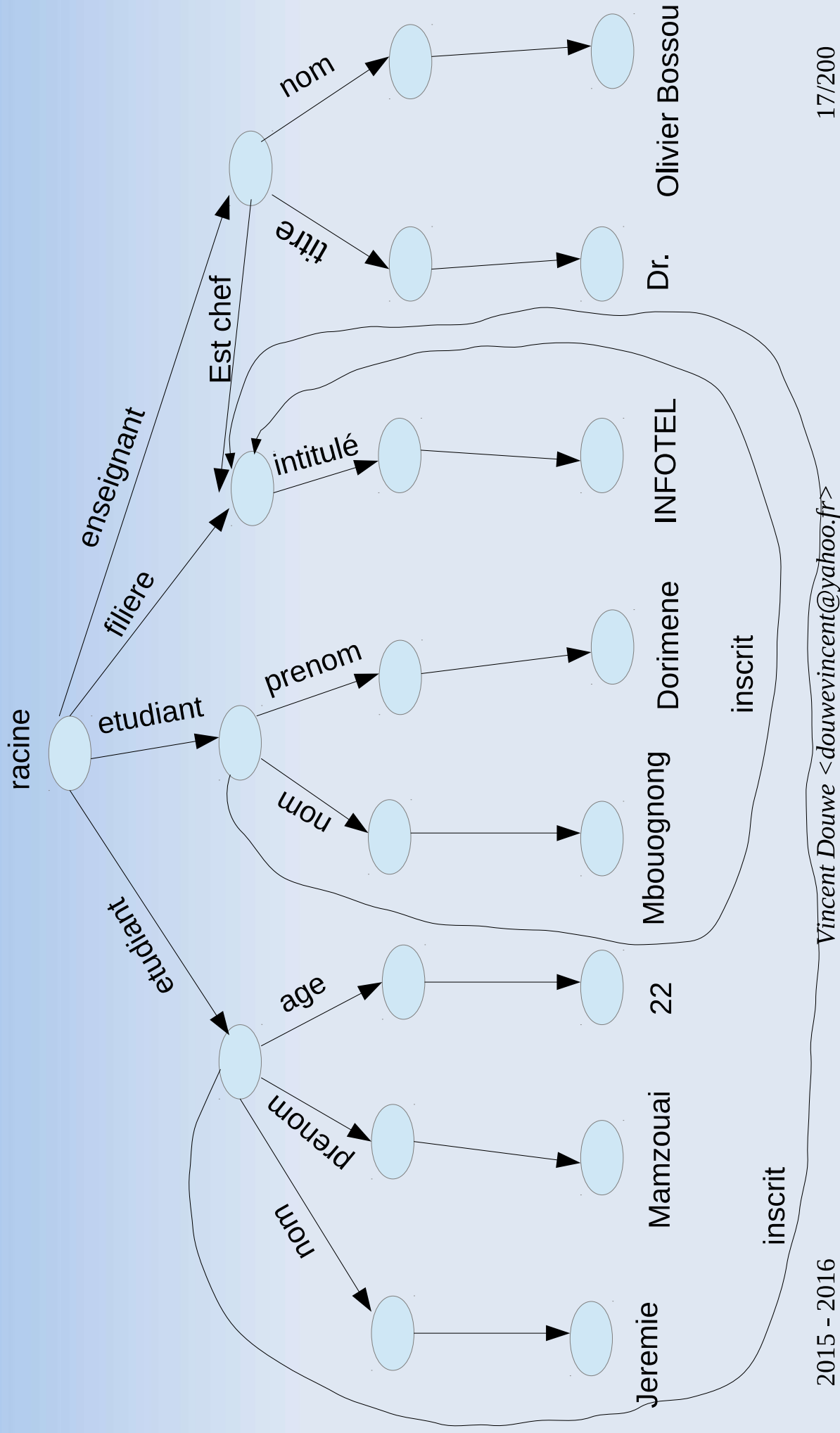
Tous les nœuds sont accessibles de la racine même comme le graphe n'est pas forcément un arbre.

Représentation

Supposons que l'on veut une base de données pour représenter les étudiants inscrits dans les différentes filières d'une école.

L'on décide matérialiser le fait qu'il existe deux étudiants appelés Jérémie Mamzouai et Mbouognong Dorimene inscrits dans une filière appelée INFOTEL qui a sa tête un enseignant appelé Olivier Bossou.

Représentation



Le rôle des arcs

Le label L d'un arc de N vers M peut jouer deux rôles différents :

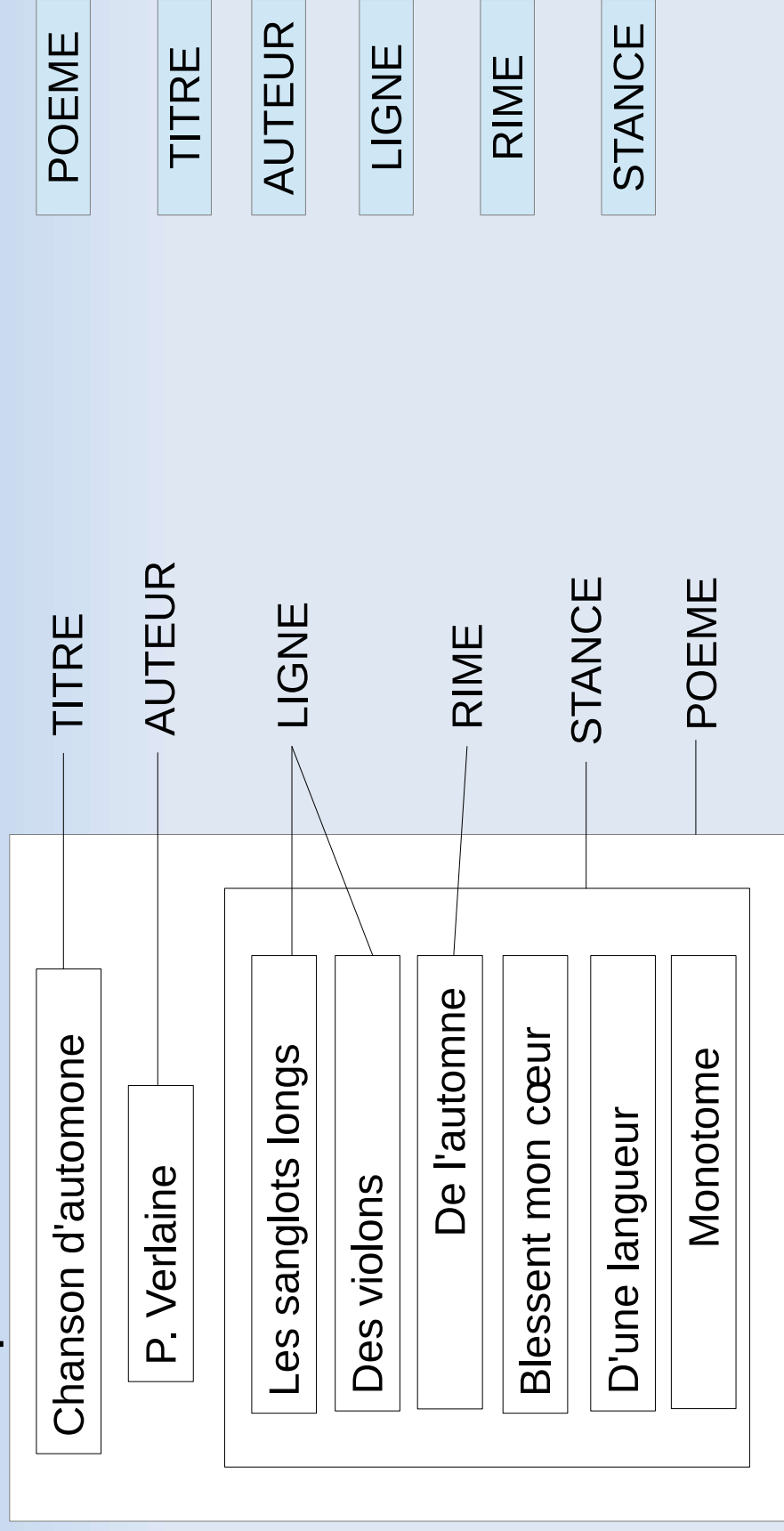
- L'on peut penser que N représente un objet ou une entité alors que M représente l'un de ses attributs. Dans ces conditions L représente le nom de l'attribut.
- L'on peut penser que N et M représentent les entités et L. Dans ces conditions, le label représente une relation entre les entités N et M.

Les standards d'échange

- SGML (norme ISO - 1986) utilisé pour les applications applications documentaires lourdes (éditeurs juridiques, publications administratives).
- HTML un langage de balises pour une mise en forme normalisée des documents échangées sur le web. C'est un langage issu de SGML
- XML est une recommandation officielle du W3C (1996-1998). Le but est de susciter et promouvoir des techniques accessibles et fiables sur le web.

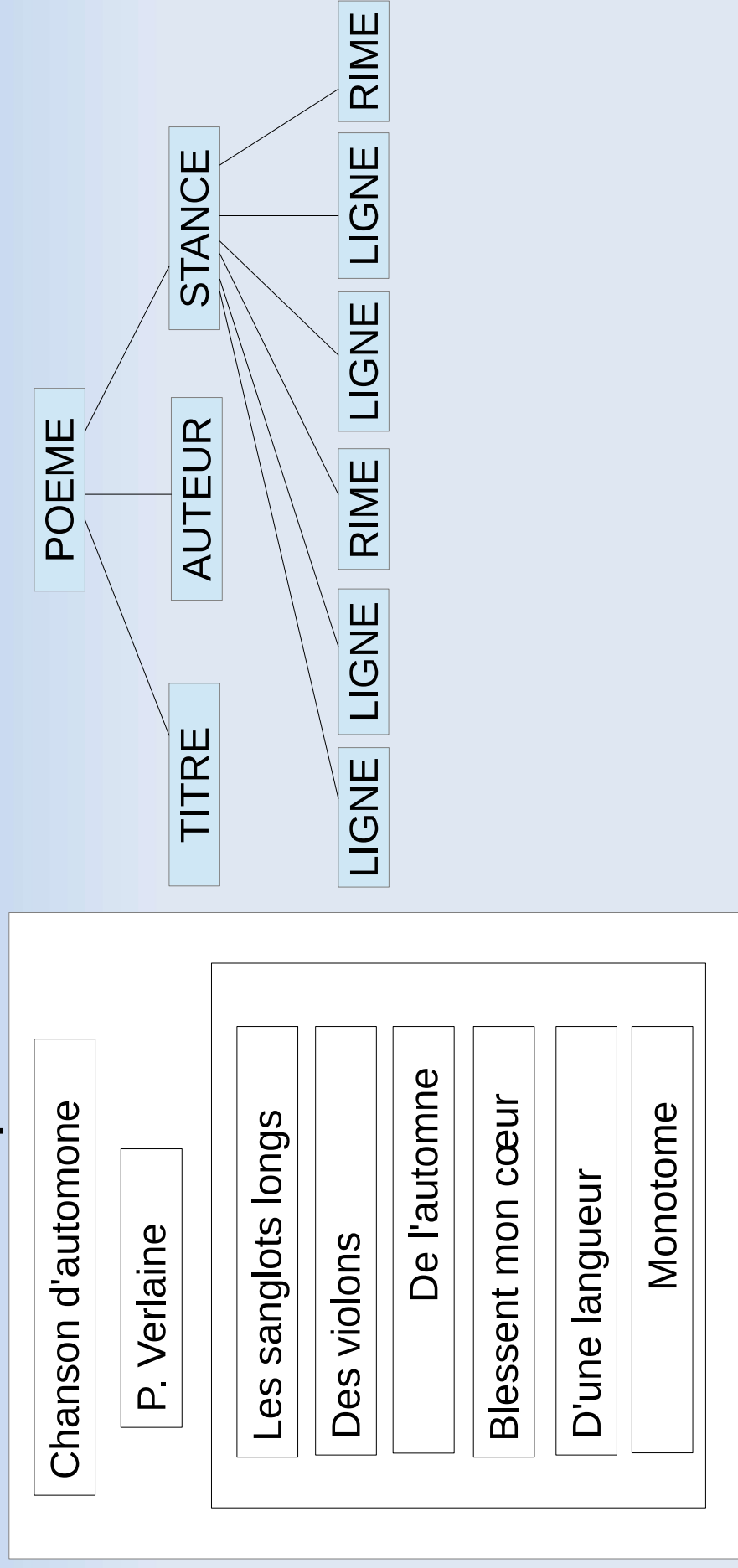
SGML

SGML (Standard Generalized Markup Language) est un langage de balises pour représenter la structure hiérarchique d'un document.



SGML

SGML est un langage de balises pour représenter la structure hiérarchique d'un document.



SGML

<POEME>

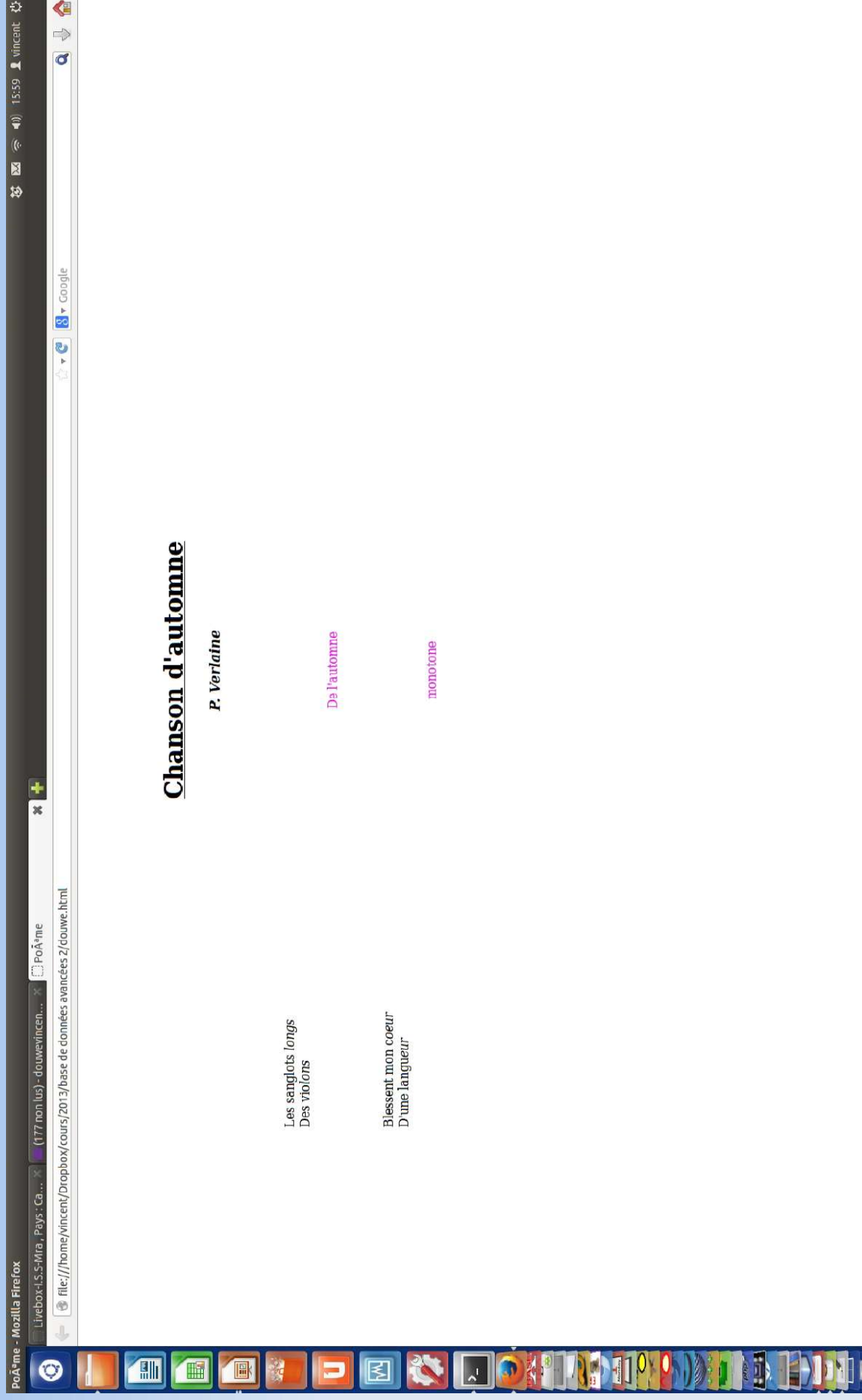
<TITRE>	Chanson d'automne	</TITRE>																		
<AUTEUR>	P. Verlaine	</AUTEUR>																		
<STANCE>	<table><tr><td><LIGNE></td><td>Les sanglots longs</td><td></LIGNE></td></tr><tr><td><LIGNE></td><td>Des violons</td><td></LIGNE></td></tr><tr><td><RIME></td><td>De l'automne</td><td></RIME></td></tr><tr><td><LIGNE></td><td>Blessent mon cœur</td><td></LIGNE></td></tr><tr><td><LIGNE></td><td>D'une langueur</td><td></LIGNE></td></tr><tr><td><RIME></td><td>Monotome</td><td></RIME></td></tr></table>	<LIGNE>	Les sanglots longs	</LIGNE>	<LIGNE>	Des violons	</LIGNE>	<RIME>	De l'automne	</RIME>	<LIGNE>	Blessent mon cœur	</LIGNE>	<LIGNE>	D'une langueur	</LIGNE>	<RIME>	Monotome	</RIME>	</STANCE>
<LIGNE>	Les sanglots longs	</LIGNE>																		
<LIGNE>	Des violons	</LIGNE>																		
<RIME>	De l'automne	</RIME>																		
<LIGNE>	Blessent mon cœur	</LIGNE>																		
<LIGNE>	D'une langueur	</LIGNE>																		
<RIME>	Monotome	</RIME>																		

</POEME>

Exemple de document HTML

```
<HTML>
<HEAD>
  <TITLE> Poème </TITLE>
  <LINK REL='stylesheet' HREF='style.css'>
</HEAD>
<BODY LEFTMARGIN="300" RIGHTMARGIN="300" TOPMARGIN="100" >
  <BASEFONT FACE="verdana" SIZE="4" COLOR="navy">
    <H1 ALIGN=CENTER><U><B>Chanson d'automne</B></U></H1>
    <H3 ALIGN=CENTER><I>P. Verlaine</I></H3>
    <BR><BR>
    <DIV ALIGN=CENTER>
      <P ALIGN=LEFT>Les sanglots <EM>longs</EM> <BR> Des vio<EM>lons</EM></P>
      <P><FONT COLOR="fuchsia" SIZE=+0.5>De l'automne</FONT></P><BR>
      <P ALIGN=LEFT>Blessent mon c<EM>oeur</EM><BR> D'une langu<EM>eur</EM></P>
      <P><FONT COLOR="fuchsia" SIZE=+0.5>monotone</FONT></P>
    </DIV>
  </BODY>
</HTML>
```

Exemple de document HTML



Le langage XML

Le but du langage XML est d'utiliser la puissance du SGML et le succès du HTML. Dans ce cas, il est :

- Un SGML simplifié et modernisé
- Un HTML générique et extensible
 - Balisage sémantique(séparation du fond de la forme)
 - Vocabulaire non figé
 - Possibilité de validation de la structure.

Le langage XML

Dans ce contexte, le langage est une méthode normalisée de représentation textuelle de données structurées hiérarchiquement apportant :

- un métalangage permettant la définition de langages de balises ;
- plusieurs formalismes de formulation d'expressions (DTD, Schémas XML) ;
- des outils de validation et de transformation.