

# Projet 2.4

---

NoDEfr2 : XML vers OWL2

Killian POULIQUEN  
Achraf TALBI  
Khalil BALGA



**01**

---

## **Le Projet**

Description et mise  
en contexte du  
projet

**02**

---

## **Formalisation**

Identification du  
problème. Solutions  
possibles. Difficultés  
rencontrées

**03**

---

## **Implémentation**

Démonstration et  
explication des de  
l'implémentation



01

# Le projet

Présentation et mise en  
contexte du projet

# Le projet 2.4 dans son contexte



## MLR

Méta-modèle MLR :  
Metadata for  
Learning Resources



## NoDEfr2

NoDEfr2 : Un modèle  
pour décrire les  
offres de formation  
du supérieur en  
France



## Projet 2.4

Formalisation du  
modèle NoDEfr2 en  
OWL2.

## Détails sur le projet 2.4

  
**Tableur**



  
**XML**



  
**XSD**



Projet 2.4







02

# Formalisation

Formalisation du problème, des  
solutions et des difficultés

# Décomposition du problème



Comprendre comment passer du paradigme MLR vers OWL.

Lister les colonnes du tableur que nous pouvons représenter en OWL2.

Passer du paradigme MLR vers le paradigme OWL.



Construire la feuille XSLT à partir d'exemples XML.

Corriger la feuille XSLT pour pouvoir l'appliquer au vrai fichier XML de l'équipe 2.2.

Corriger la feuille XSLT pour que le fichier turtle soit syntaxiquement validé par le site du W3C.

# Solutions possibles



## Convertisseurs en ligne

### Avantages :

- Solution clé en main.
- Fichier en sortie valide.

### Inconvénients :

- Fichier XML d'entrée doit respecter la syntaxe XML/RDF.
- Peu paramétrable.
- Pas très disponible pour une conversion en OWL2.



## Programme Python ou autre langage

### Avantages :

- Bonnes compétences en Python.
- Module Python disponible pour manipuler des arbres XML.

### Inconvénients :

- Python n'est pas spécialisé dans le traitement des données XML.
- Type de solution non abordée dans le cadre de ce cours.



## Feuille XSLT

### Avantages :

- Langage spécialisé pour traiter des documents XML.
- Compilation simple grâce à un processeur XSLT tel que Saxon.
- Débuggage simple grâce aux messages d'erreurs produite par les processeurs XSLT.



# Difficultés rencontrées



La mise en page TTL



Transformation XML - TTL



XSLT : langage à balise orienté fonctionnel



03

# L'implémentation

Démonstration et explication de  
l'implémentation

# Démonstration

---

Utilisation du processeur XSLT **Saxon**

# Explication



Gestion des **Classes**



Gestion de *"l'héritage"*



Gestion des **Relations**



Gestion des **Restrictions**

# THANKS!

---

Do you have any questions?  
addyouremail@freepik.com  
+91 620 421 838  
yourcompany.com



---

CREDITS: This presentation template was created by Slidesgo, including icons by Flaticon, and infographics & images by Freepik.

**Please keep this slide for attribution.**