

## Xmlia : Editeur Xml

BOIVIN Benoit  
LE PHILIPPE Noé  
KEGBA-SANGO-SANGO Ulrich-Chancelin  
WOUTERS Stéphane

25 mai 2014

# Sommaire

- 1 Présentation du projet
  - Editeur XML
  - Cahier des charges
  - L'existant
- 2 Organisation du projet
- 3 Développement
- 4 Résultat
- 5 Conclusion

# Éditeur XML

- Le langage XML

# Éditeur XML

- Le langage XML
- Les besoins d'un éditeur

# Cahier des charges

- Deux vues
  - Vue textuelle

# Cahier des charges

- Deux vues
  - Vue textuelle
    - Coloration syntaxique

# Cahier des charges

- Deux vues
  - Vue textuelle
    - Coloration syntaxique
    - Indentation automatique

# Cahier des charges

- Deux vues
  - Vue textuelle
    - Coloration syntaxique
    - Indentation automatique
    - Auto complétion



# Cahier des charges

- Deux vues
  - Vue textuelle
    - Coloration syntaxique
    - Indentation automatique
    - Auto complétion
  - Vue arborescente

# Cahier des charges

- Deux vues
  - Vue textuelle
    - Coloration syntaxique
    - Indentation automatique
    - Auto complétion
  - Vue arborescente
    - Arbre hiérarchique

# Cahier des charges

- Deux vues
  - Vue textuelle
    - Coloration syntaxique
    - Indentation automatique
    - Auto complétion
  - Vue arborescente
    - Arbre hiérarchique
    - Actions

# Cahier des charges

- Deux vues
  - Vue textuelle
    - Coloration syntaxique
    - Indentation automatique
    - Auto complétion
  - Vue arborescente
    - Arbre hiérarchique
    - Actions
- Validation syntaxique

# Cahier des charges

- Deux vues
  - Vue textuelle
    - Coloration syntaxique
    - Indentation automatique
    - Auto complétion
  - Vue arborescente
    - Arbre hiérarchique
    - Actions
- Validation syntaxique
- Liaison d'un schéma

# L'existant

- De nombreux éditeurs existants

# L'existant

- De nombreux éditeurs existants
  - De nombreuses options

# L'existant

- De nombreux éditeurs existants
  - De nombreuses options
  - Payants ou gratuits



# L'existant

- De nombreux éditeurs existants
  - De nombreuses options
  - Payants ou gratuits
  - Un exemple : oXygen

# Sommaire

- 1 Présentation du projet
- 2 Organisation du projet
  - Choix de développement
  - Fonctionnement du groupe
- 3 Développement
- 4 Résultat
- 5 Conclusion

# Choix de développement

- Langage et bibliothèques
  - C++
  - Qt

# Choix de développement

- Langage et bibliothèques
  - C++
  - Qt
- Gestionnaire de projet
  - Git

# Choix de développement

- Langage et bibliothèques
  - C++
  - Qt
- Gestionnaire de projet
  - Git
  - Service Bitbucket

# Fonctionnement du groupe

- Communication

# Fonctionnement du groupe

- Communication
  - Réunions

# Fonctionnement du groupe

- Communication
  - Réunions
  - Outils



# Fonctionnement du groupe

- Communication
  - Réunions
  - Outils
    - Google Hangout

# Fonctionnement du groupe

- Communication
  - Réunions
  - Outils
    - Google Hangout
    - Google Drive

# Fonctionnement du groupe

- Communication
  - Réunions
  - Outils
    - Google Hangout
    - Google Drive
    - BitBucket

# Fonctionnement du groupe

- Communication
  - Réunions
  - Outils
    - Google Hangout
    - Google Drive
    - BitBucket
- Répartition des tâches
  - En fonction des compétences de chacun
  - Système de ticket

# Sommaire

- 1 Présentation du projet
- 2 Organisation du projet
- 3 Développement**
  - Qt
  - Architecture
  - Les problèmes rencontrés
- 4 Résultat
- 5 Conclusion

# Présentation de Qt

Qu'est ce que Qt

Api orientée objet d'interface graphique

# Présentation de Qt

Qu'est ce que Qt

Api orientée objet d'interface graphique

Fonctionnement général

Un système exhaustif de widget

# Présentation de Qt

Qu'est ce que Qt

Api orientée objet d'interface graphique

Fonctionnement général

Un système exhaustif de widget

De nombreux modules et outils

Un grand nombre fonctionnalités déjà présentes



# Modèle MVC de Qt

Le modèle

Donné par le développeur

# Modèle MVC de Qt

Le modèle

Donné par le développeur

La vue

Gérée par Qt

# Modèle MVC de Qt

Le modèle

Donné par le développeur

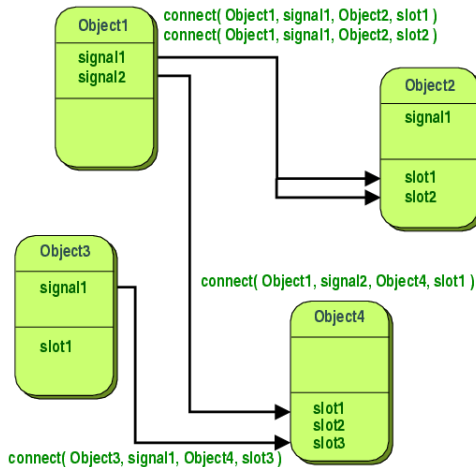
La vue

Gérée par Qt

Le contrôleur

Un mélange des deux

# Signal/Slot



# Conception

Quelle architecture ?

La contrainte du modèle de widget

# Conception

Quelle architecture ?

La contrainte du modèle de widget

Choisir nos Widgets

Intimement lié à la connaissance de Qt

# Conception

Quelle architecture ?

La contrainte du modèle de widget

Choisir nos Widgets

Intimement lié à la connaissance de Qt

Les faire communiquer

Utilité du modèle

# Conception

Quelle architecture ?

La contrainte du modèle de widget

Choisir nos Widgets

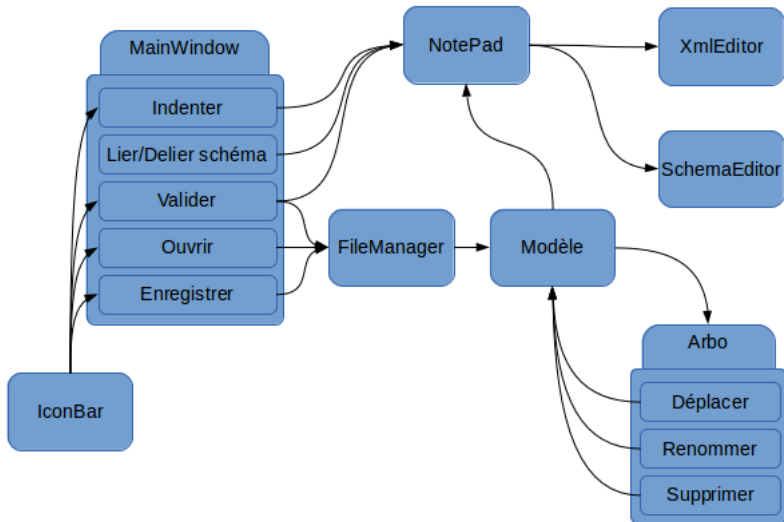
Intimement lié à la connaissance de Qt

Les faire communiquer

Utilité du modèle



# Communication entre nos composants



# La vue arborescente

L'affichage

Modifier le QDomDocument

# La vue arborescente

L'affichage

Modifier le QDomDocument

Le glisser-déposer

Les signaux de Qt

# L'éditeur de texte

Se repérer dans le texte

Correspondance balise/emplacement

# L'éditeur de texte

Se repérer dans le texte

Correspondance balise/emplacement

Manipuler le texte

Rester cohérent lors de changements dans l'arborescence

# Sommaire

- 1 Présentation du projet
- 2 Organisation du projet
- 3 Développement
- 4 Résultat**
- 5 Conclusion

# Résultat

The screenshot shows an XML editor interface. On the left, a tree view displays the structure of the XML file 'schema\_example.xml'. The tree is expanded to show the 'shiporder' element, which contains 'orderperson', 'shipto', and 'item' elements. The 'shipto' element is further expanded to show 'name', 'address', 'city', and 'country'. The 'item' element is also expanded to show 'title', 'note', 'quantity', and 'price'.

The main editor area displays the XML content, which is a combination of a schema and an instance. The schema part defines the structure and data types of the elements. The instance part provides the actual data values for the elements. The XML is as follows:

```

1 <?xml version="1.0" encoding="UTF-8"?>
2
3 <shiporder orderid="889923">
4   xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
5   xsi:noNamespaceSchemaLocation="schema.xsd">
6     <orderperson>John Smith</orderperson>
7     <shipto>
8       <name>Ola Nordmann</name>
9       <address>Langgt 23</address>
10      <city>4000 Stavanger</city>
11      <country>Norway</country>
12    </shipto>
13    <item>
14      <title>Empire Burlesque</title>
15      <note>Special Edition</note>
16      <quantity>1</quantity>
17      <price>10.90</price>
18    </item>
19    <item>
20      <title>Hide your heart</title>
21      <quantity>1</quantity>
22      <price>9.90</price>
23    </item>
24  </shiporder>
  
```

At the bottom of the editor, a status bar shows the results of various validation checks:

- Opened file (XML) /home/noe/dev/QL/Xmlia/Projet/schema\_example.xml
- Syntaxe XML valide
- Opened file (DTD) /home/noe/dev/QL/Xmlia/Projet/schema.xsd
- Opened file (DTD) /home/noe/dev/QL/Xmlia/Projet/schema.xsd
- Schema XSD valide
- Semantique XML valide

# Sommaire

- 1 Présentation du projet
- 2 Organisation du projet
- 3 Développement
- 4 Résultat
- 5 Conclusion**



# Conclusion

- Développement sous Qt
- Travail en groupe
- En bref