

Validation XML

Guy Lapalme

Intra
question courts
HTML et validation
XML Schema

But

- Est-ce que les éléments sont imbriqués selon ce qui est attendu
- Est-ce que le contenu des balises correspond à des valeurs attendues

Moyen

- Définition d'un schéma
 - DTD
 - XML Schema
 - Relax NG Compact (RNC)
- Confrontation entre le fichier XML et le Schéma

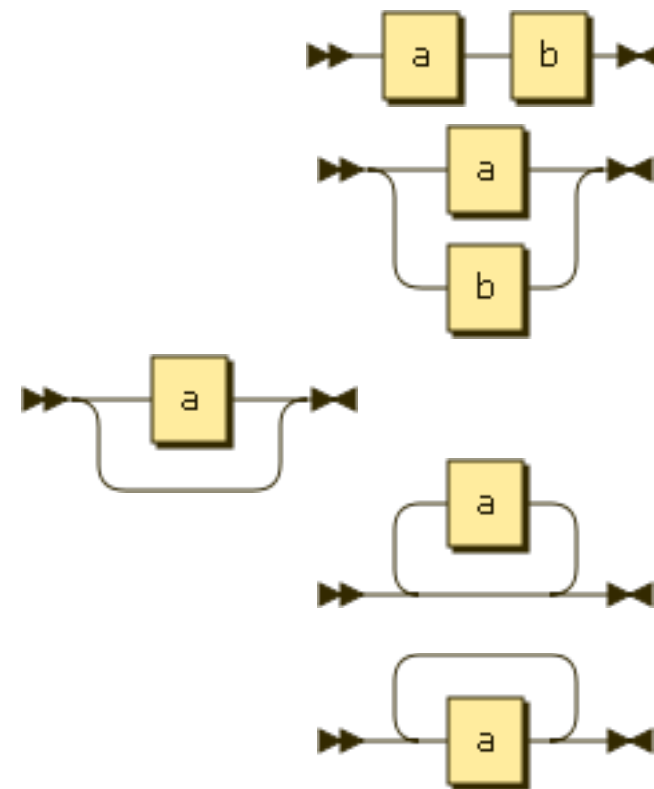
different et avantage l'un par rapport l'autre?!

Définition d'un schéma RNC

- suite de patrons nommés
 - `id = pattern`
- un des patrons est associé à `start`
- patrons
 - `element nom { patron }`
 - `attribute nom { patron }`
- combinaisons de patrons

Patron

- **element** *nom* { *patron* }
- **attribute** *nom* { *patron* }
- **mixed** *nom* { *patron* }
- combinaison de patrons (*a*,*b*)
 - séquence : *a* , *b*
 - alternative : *a* | *b*
 - groupement: (*patron*)
 - optionnel : *a* ?
 - répétition
 - 0 ou plus : *a* *
 - 1 ou plus : *a* +



Types simples

- text
- XMLSchema datatypes
 - xsd:integer
 - xsd:gYear
 - xsd:anyURI
 - ...
- Contraintes / Facettes janbeh/ side
 - minInclusive, maxInclusive
 - minLength
 - pattern
 - ...

```

<auteur id="esope">
  <nom>Ésope</nom>
  <mort>-0560</mort>
  <prose>
    <head>La Cigale et les Fourmis</head>
    <p>C'était l'hiver [...] <q>Pourquoi</q> n'as-tu pas
      fait de provisions pendant l'été ? </p>
    <p>Je n'étais pas oisive, dit-elle, je chantais en artiste.</p>
    <p>Ah l'été, tu étais musicienne repartirent les fourmis en riant ;
      en hiver fais-toi danseuse.</p>
  </prose>
</auteur>

```

↑ XML valide p/r ↓ schéma RNC

```

datatypes xsd = "http://www.w3.org/2001/XMLSchema-datatypes"

```

important en examen

xsd ou xs qui exige d'utiliser le meme code par la suite

```

start = element auteur {
  attribute id{xsd:ID},
  nom, mort, prose+
}

```

ils sont seulement des noms pour
referer sur les elements en bas

```

nom = element nom {text}
mort = element mort {xsd:gYear}
prose = element prose {
  element head{text},
  p+
}

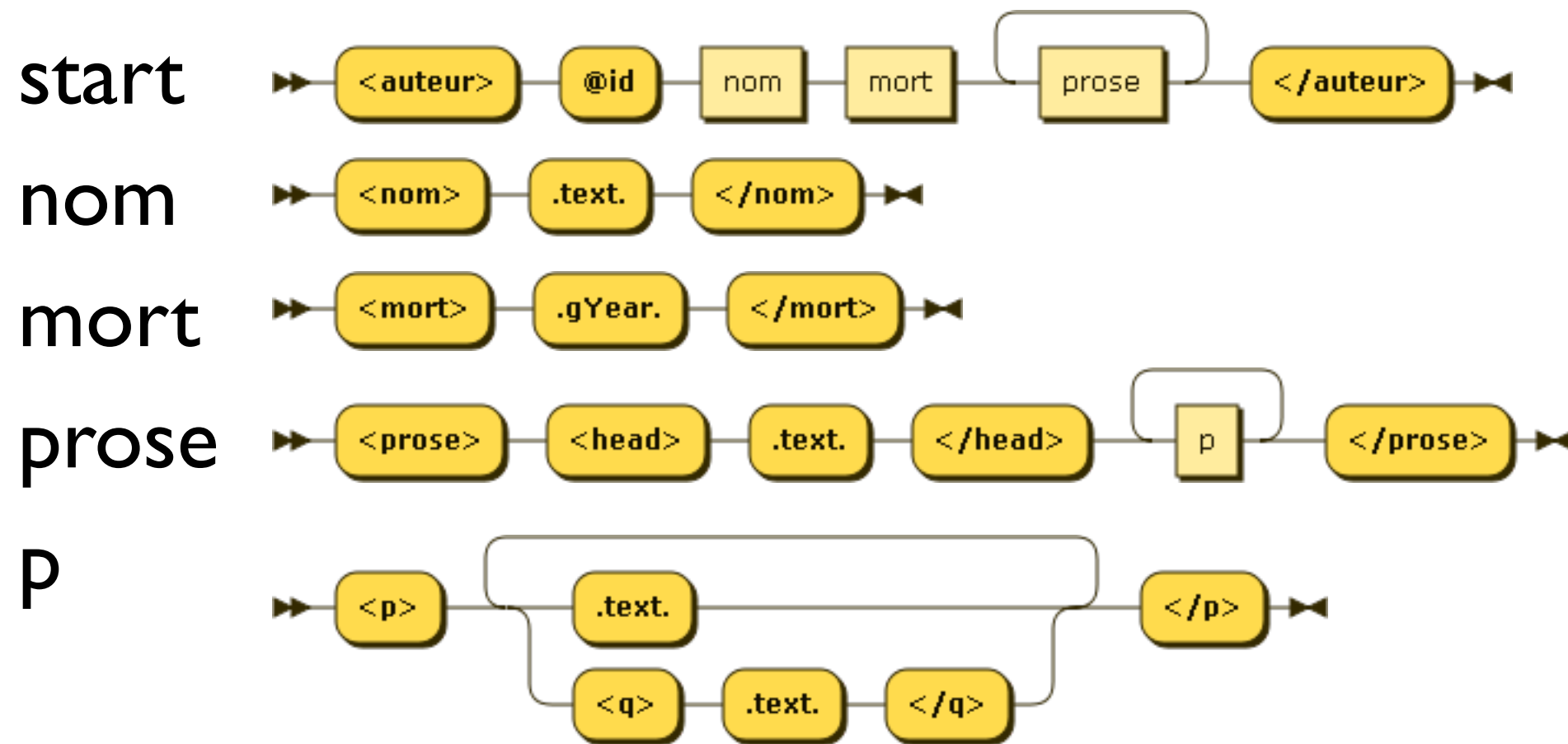
```

mixed est contient soit un element soit un texte

```

p = element p {mixed {element q{text}*}}

```



↑ diagramme vs ↓ schéma RNC

```

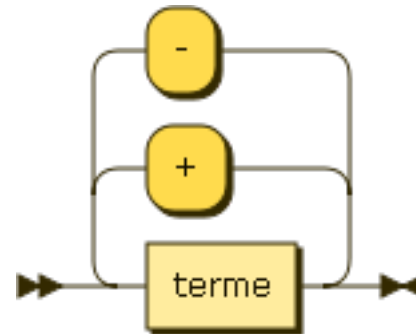
start = element auteur {
    attribute id{xsd:ID},
    nom, mort, prose+
}
nom = element nom {text}
mort = element mort {xsd:gYear}
prose = element prose {
    element head{text},
    p+
}
p = element p {mixed {element q{text}*}}
  
```

voici deux facons
declarer
d'elements!

regarde les types
personnalisés dans les
exemples

Diagrammes récursifs

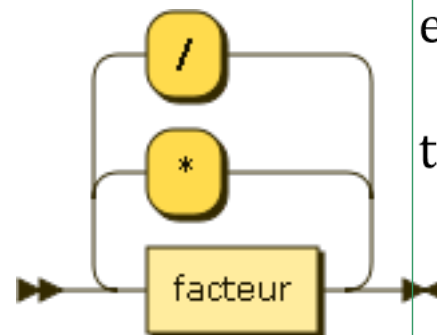
- **expression**



info = nom, prenom

student = element student {info, ...}

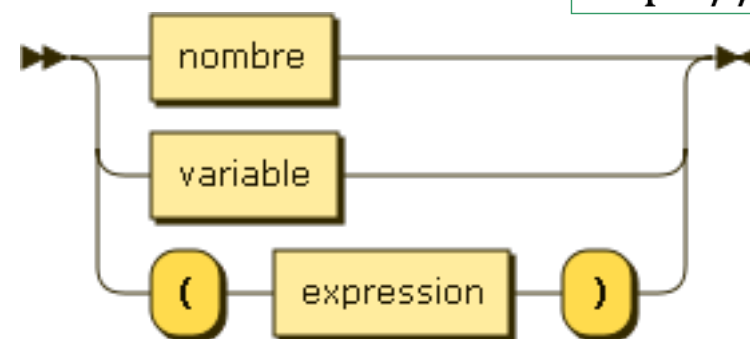
- **terme**



element tel{telCA}

telCA = xs:string{pattern="[0-9]{3}-[0-9]{3}-[0-9]{4}"}
<https://regex101.com/>

- **facteur**



pattern d'une expression reguliere: xs: string{pattern="[A-Z][0-9][...]"}

xs:"type"{pattern=""}