XPath
Exemples détaillés

INU3011 Documents structurés

Le document

- Le document utilisé dans les exemples est

  200-Ex-XPath/doc-simple-dense.xml
- Ce document n'est **pas indenté**
  - Évite de nombreux nœuds textuels "blancs"
- Il est montré à la diapo suivante en version "aérée", i.e. indentée, pour être plus lisible

  200-Ex-XPath/doc-simple.xml

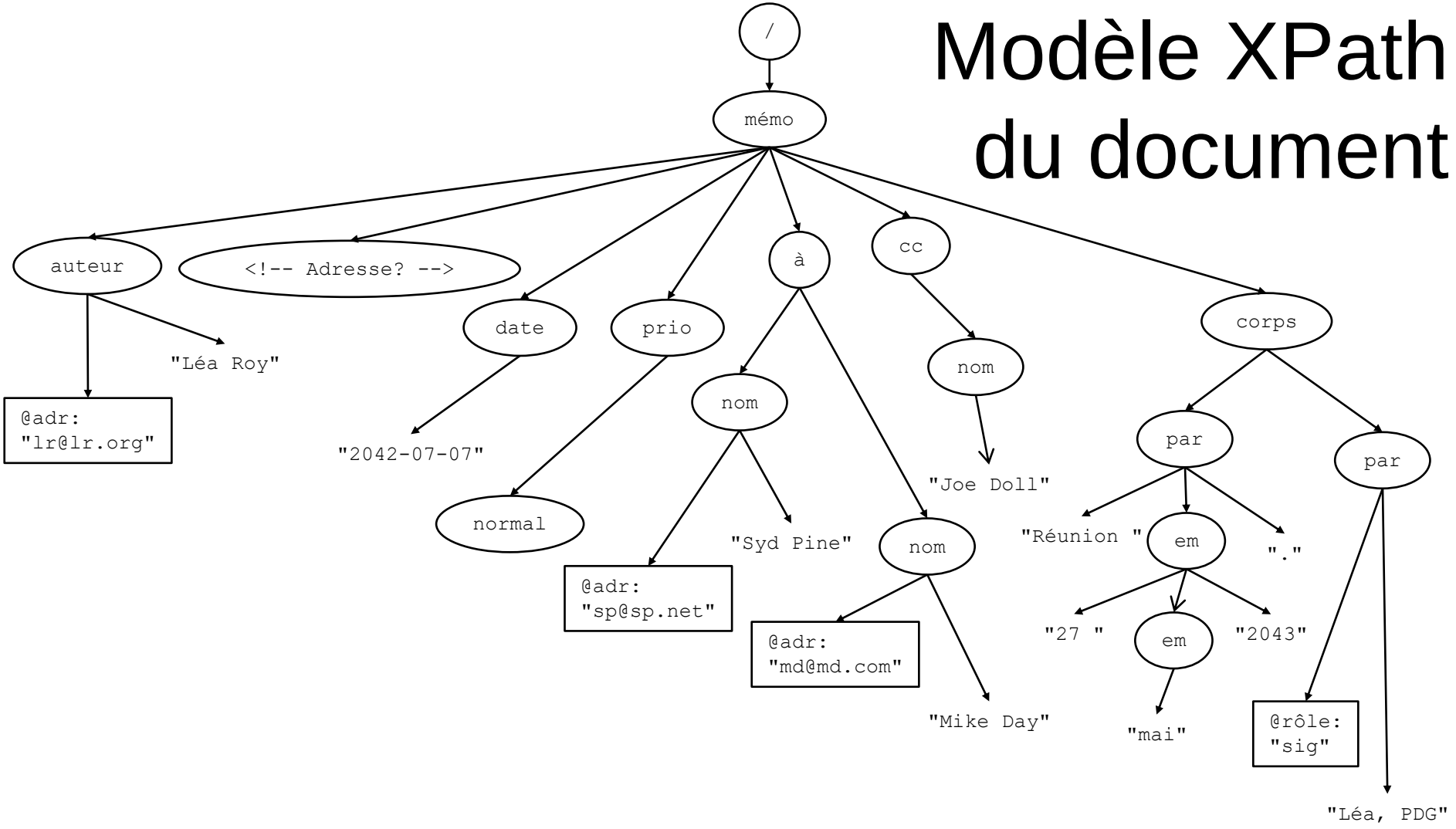
200-Ex-XPath/doc-simple.xml

Le document

(version "aérée")

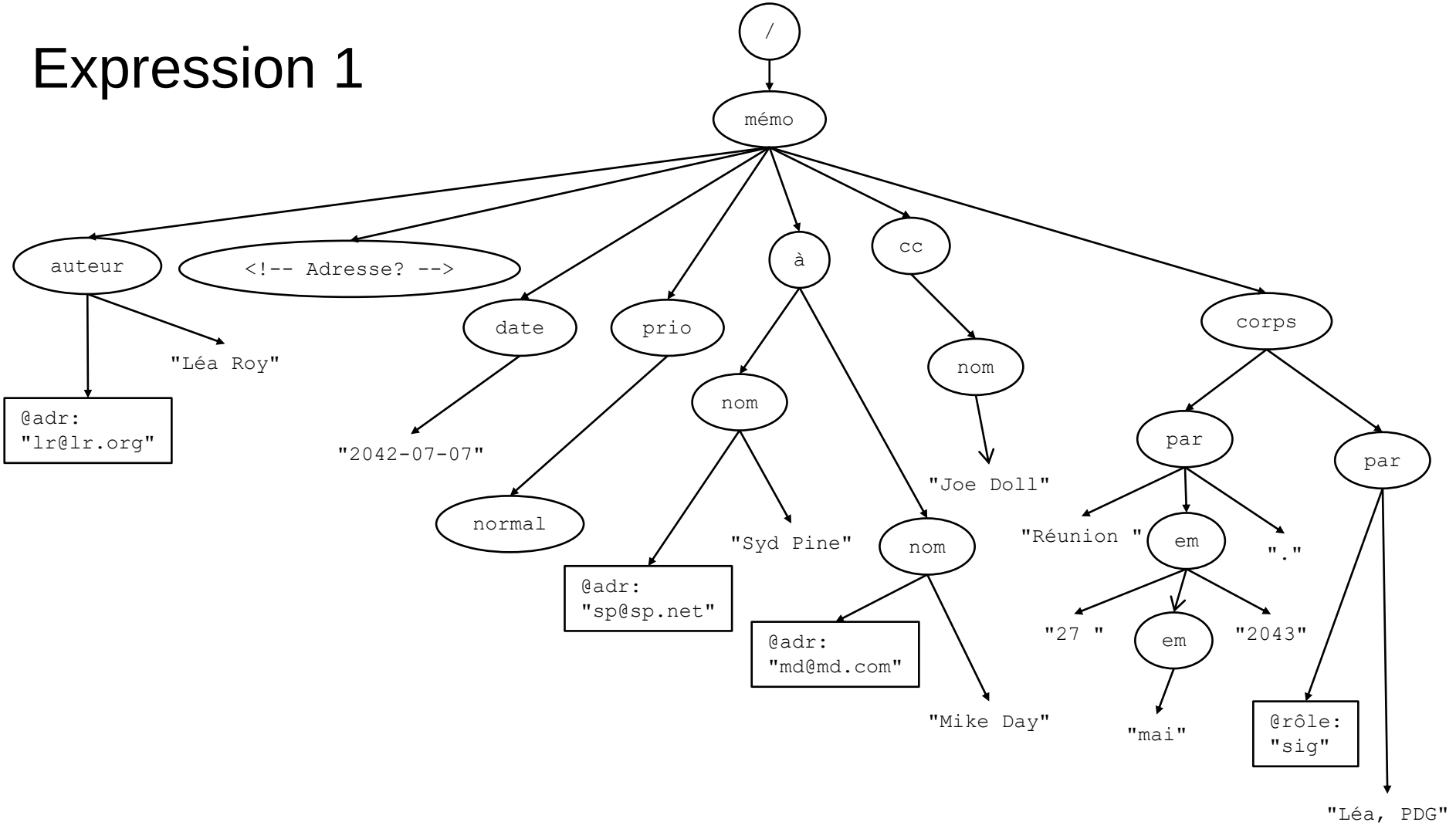
```
<mémo>
  <auteur adr="lr@lr.org">Léa Roy</auteur>
  <!-- Adresse? -->
  <date>2042-07-07</date>
  <prio><normal /></prio>
  <à>
    <nom adr="sp@sp.net">Syd Pine</nom>
    <nom adr="md@md.com">Mike Day</nom>
  </à>
  <cc>
    <nom>Joe Doll</nom>
  </cc>
  <corps>
    <par>Réunion <em>27 <em>mai</em> 2043</em>.</par>
    <par rôle="sig">Léa, PDG</par>
  </corps>
</mémo>
```

Modèle XPath du document



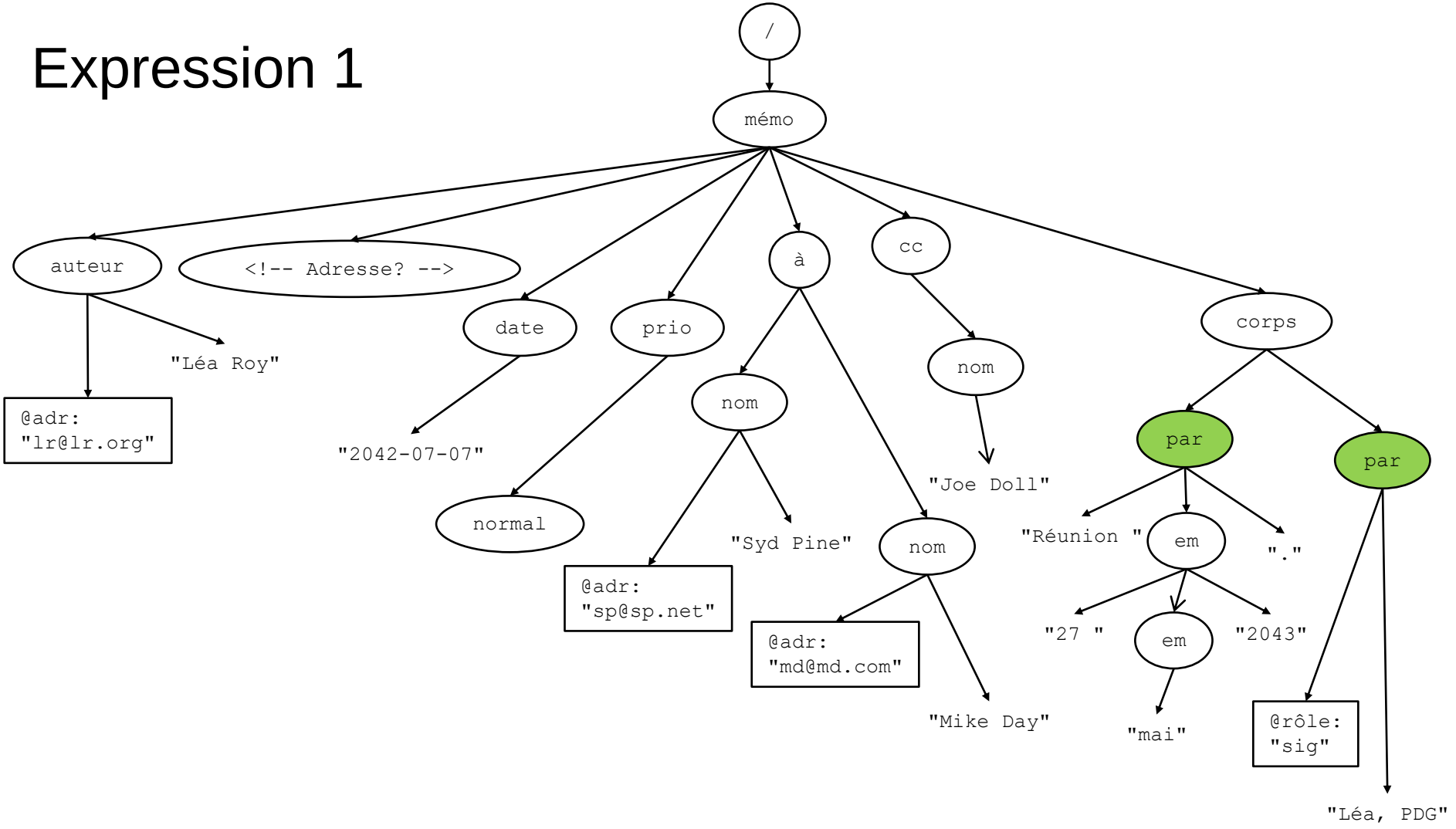
N.B.: Les nœuds textuels sont montrés avec des guillemets, plutôt qu'avec les blancs représentés par ¤

Expression 1



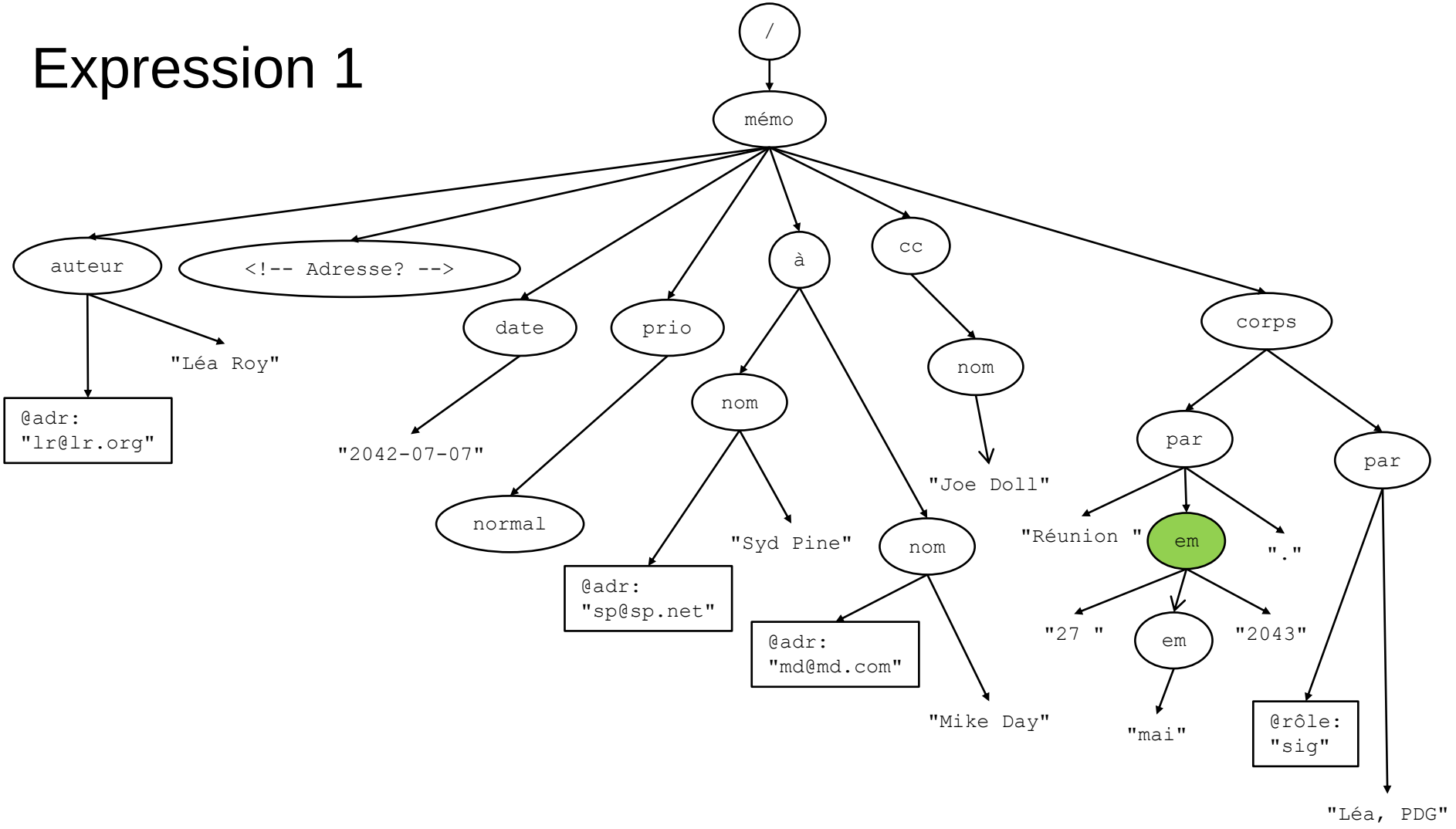
Expression: `//par/*`

Expression 1



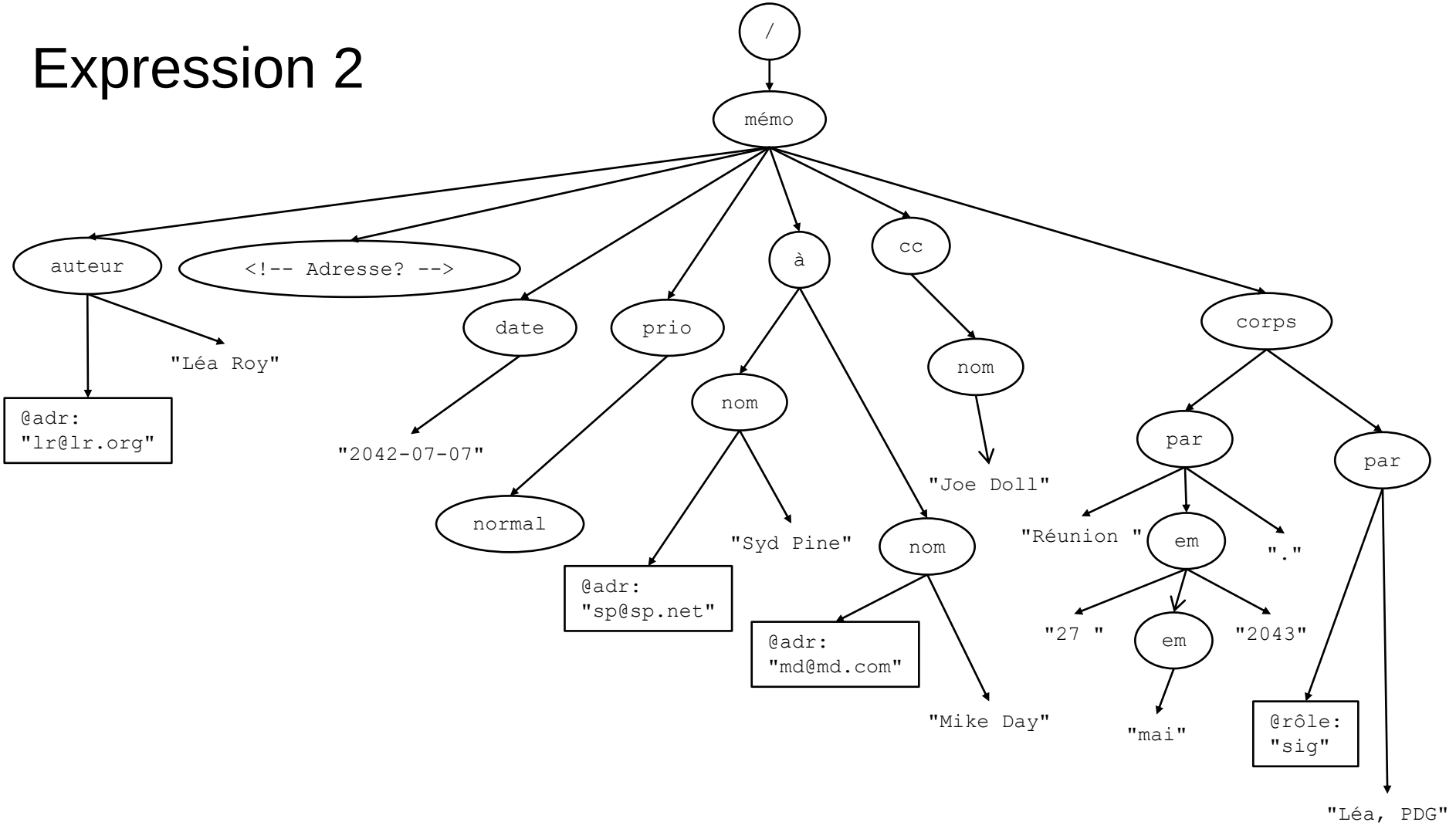
Expression: `//par/*`

Expression 1



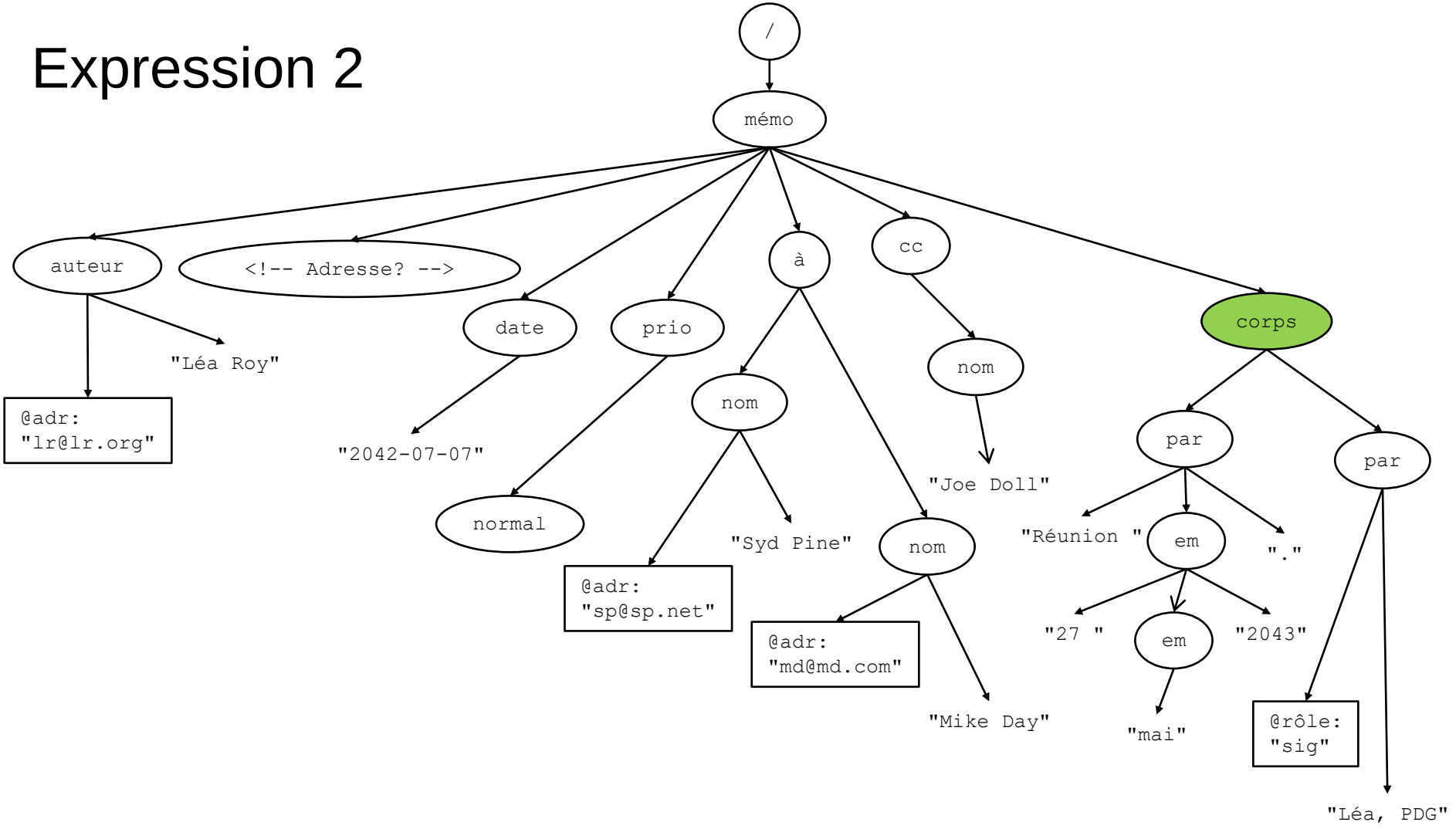
Expression: `//par/*`

Expression 2



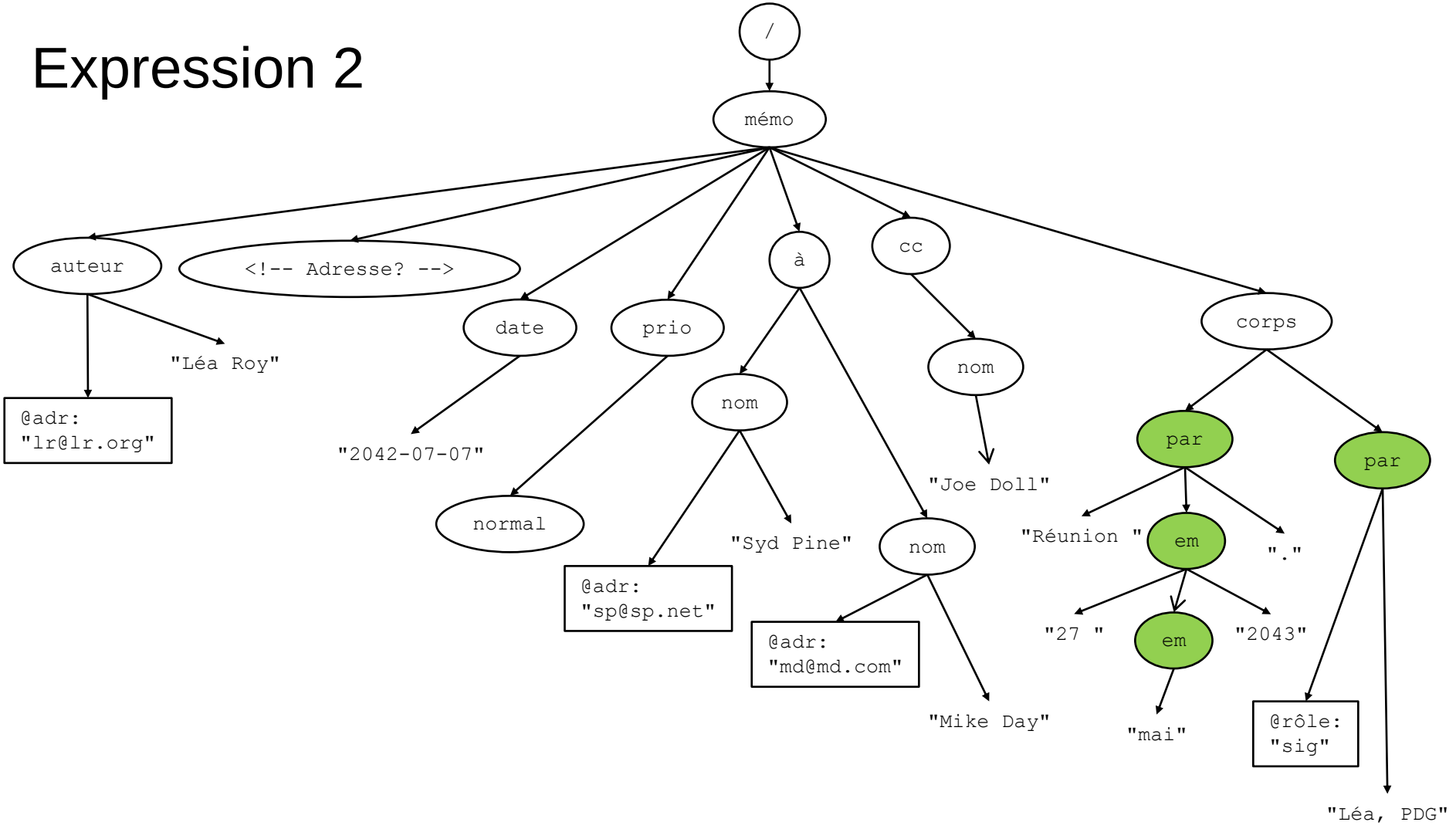
Expression: `//corps//*`

Expression 2



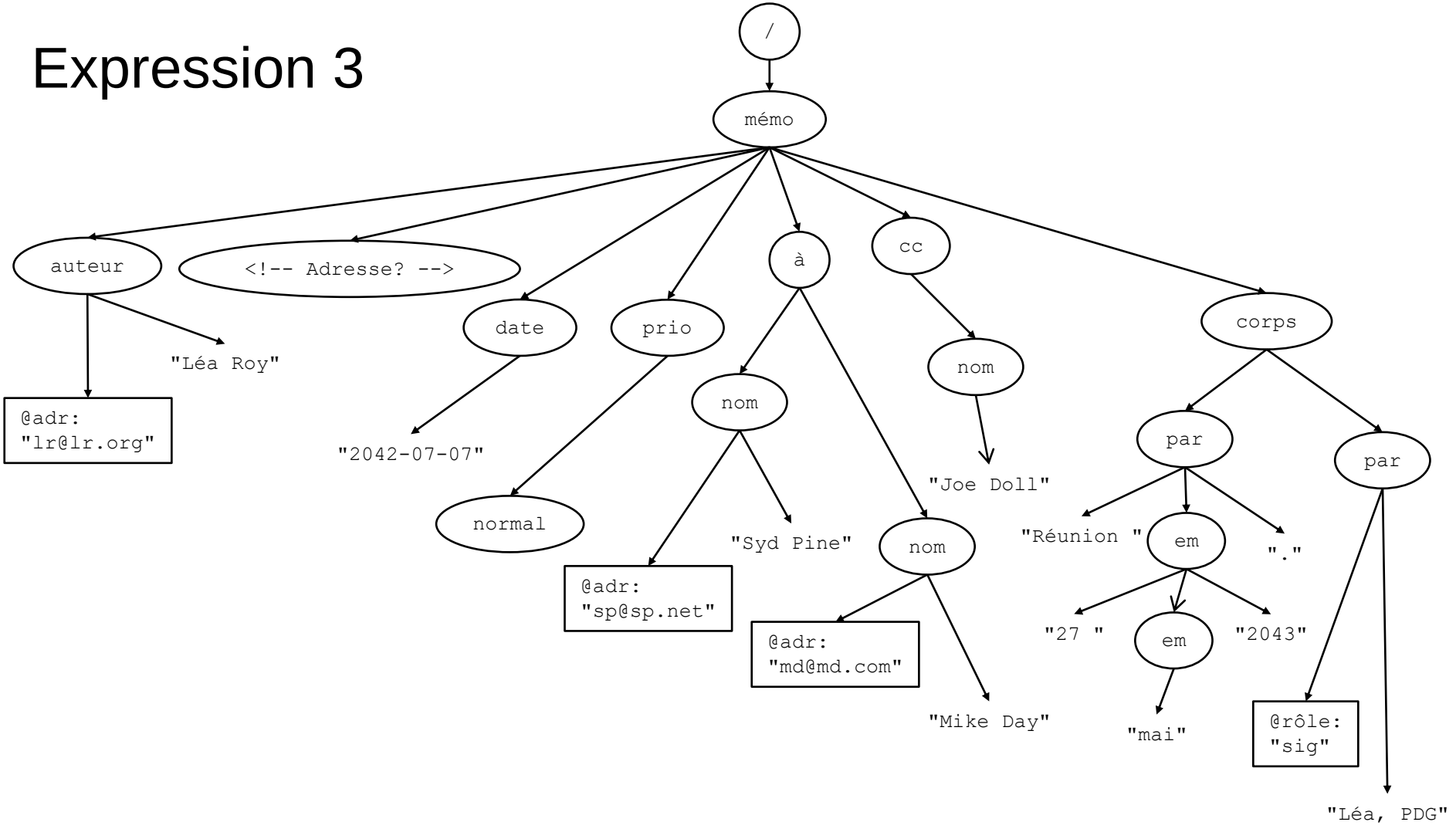
Expression: `//corps//*`

Expression 2



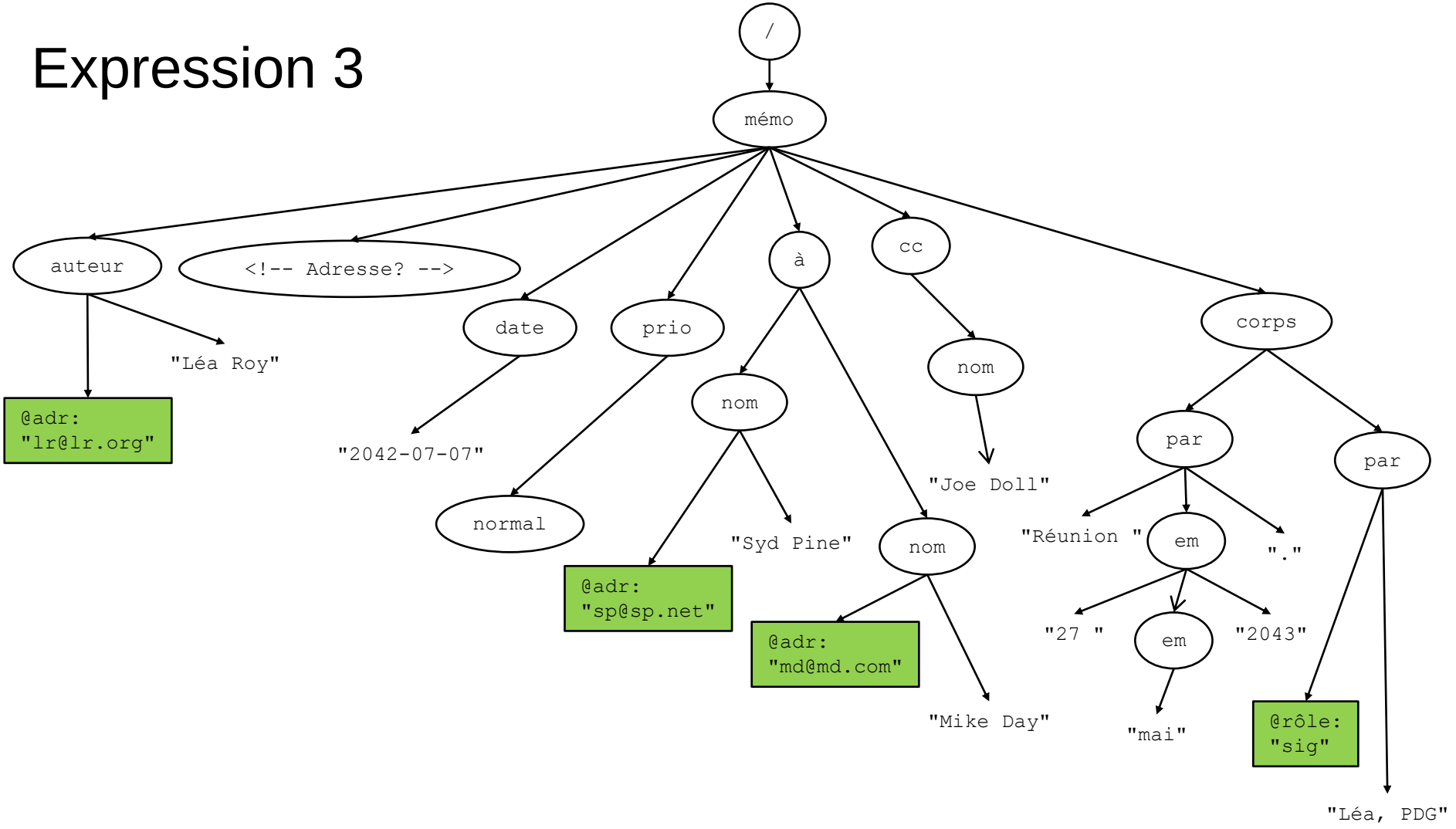
Expression: `//corps//*`

Expression 3



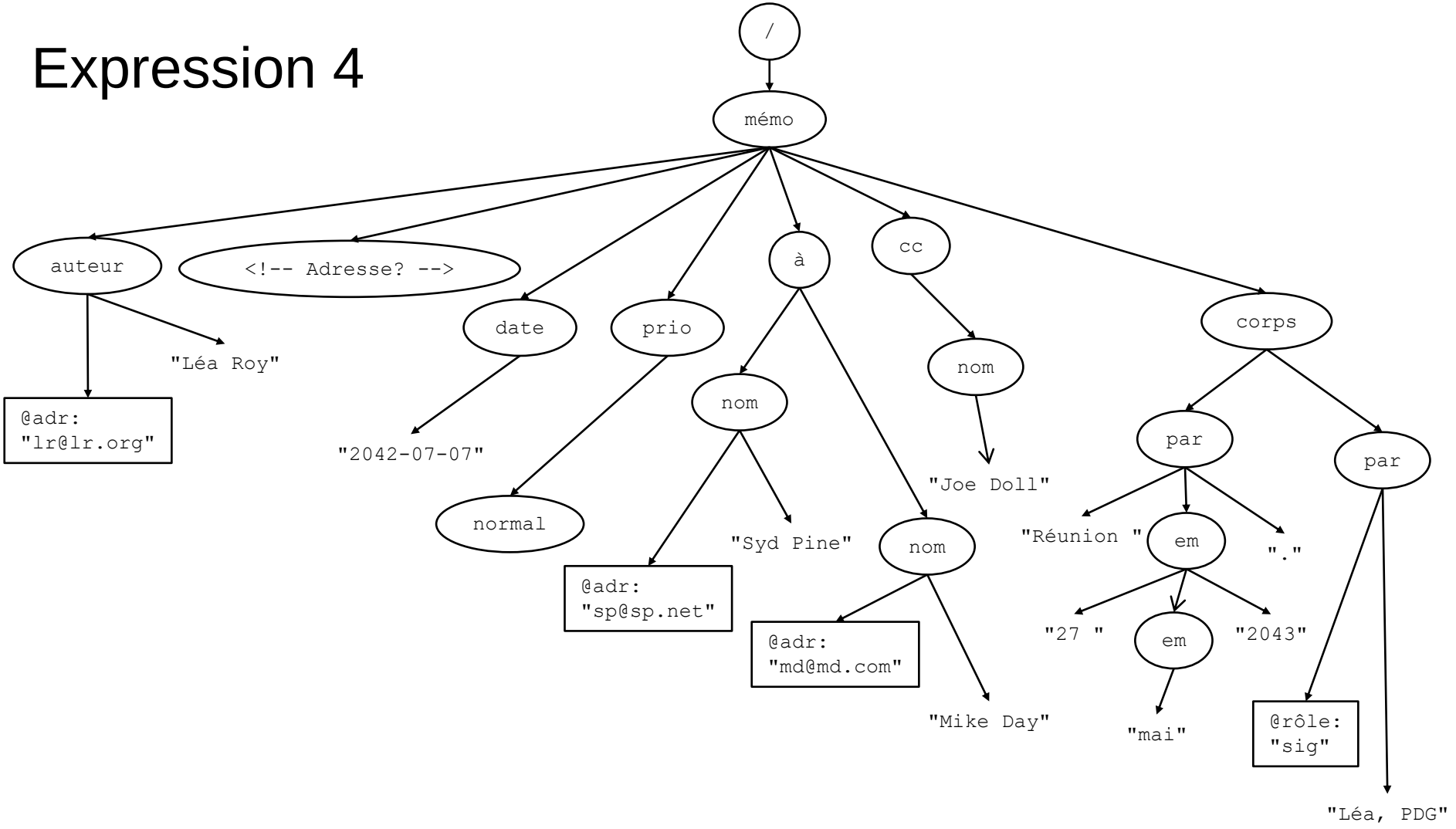
Expression: `//@*`

Expression 3



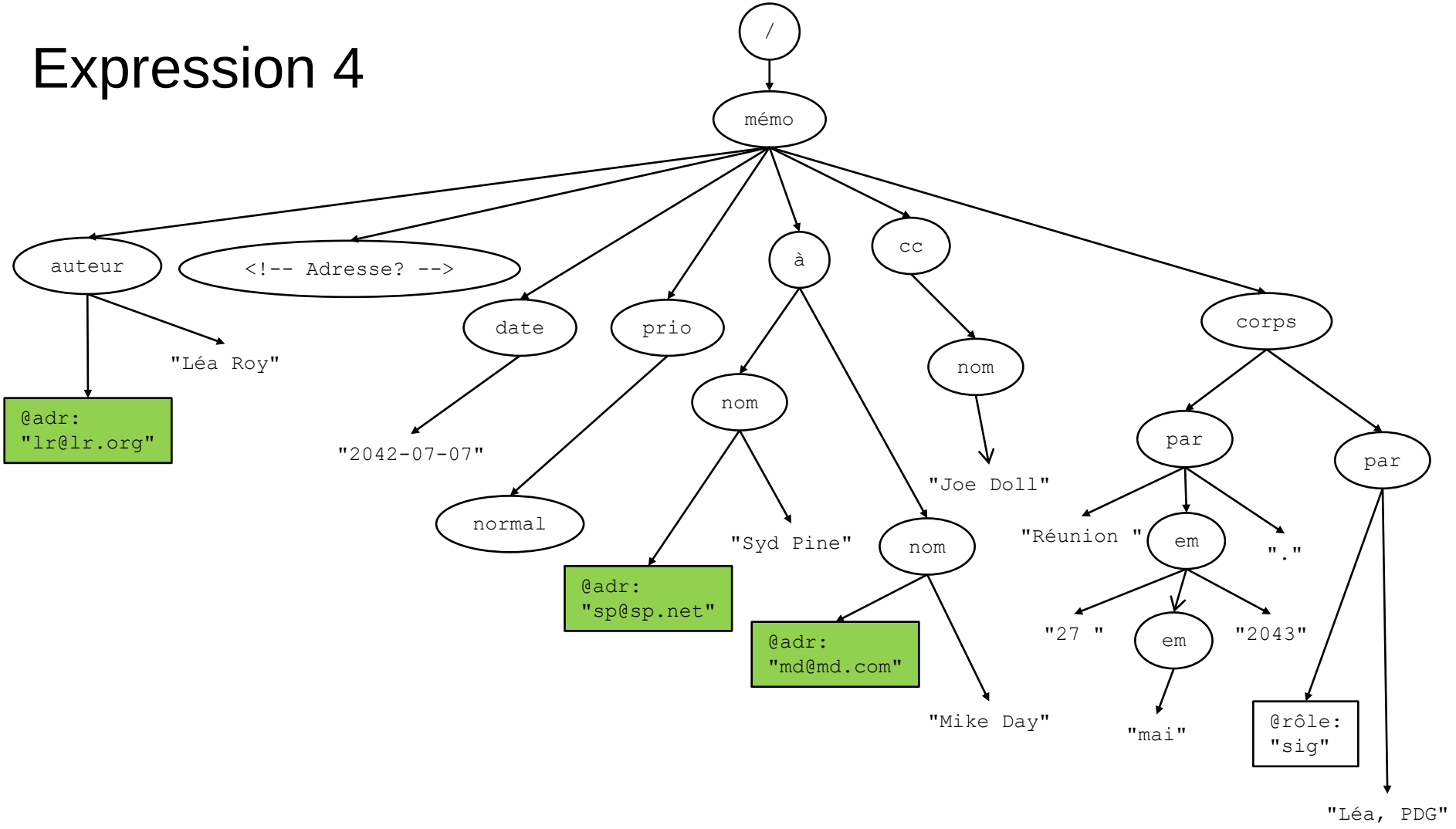
Expression: `//@*`

Expression 4



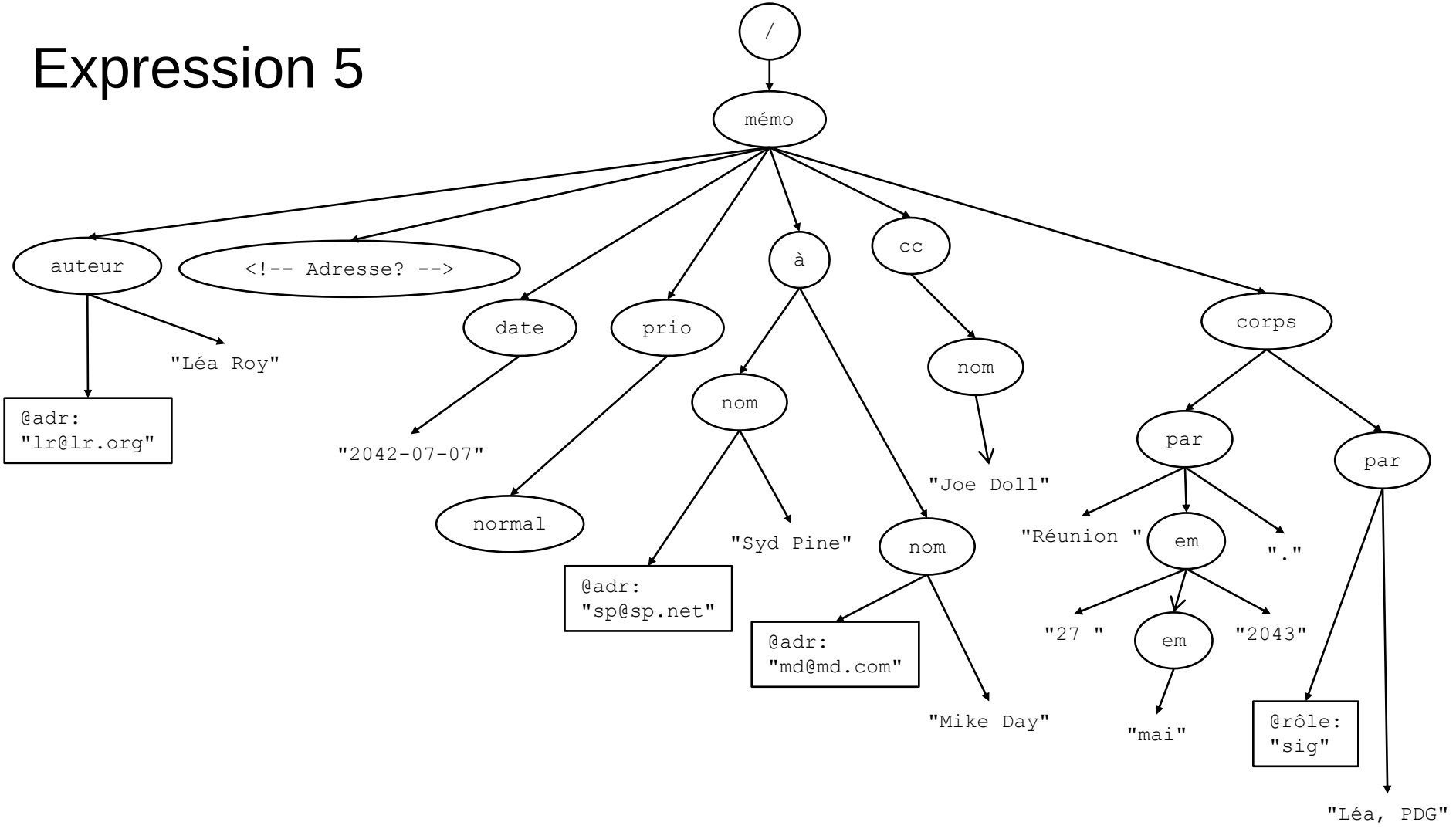
Expression: `//@adr`

Expression 4



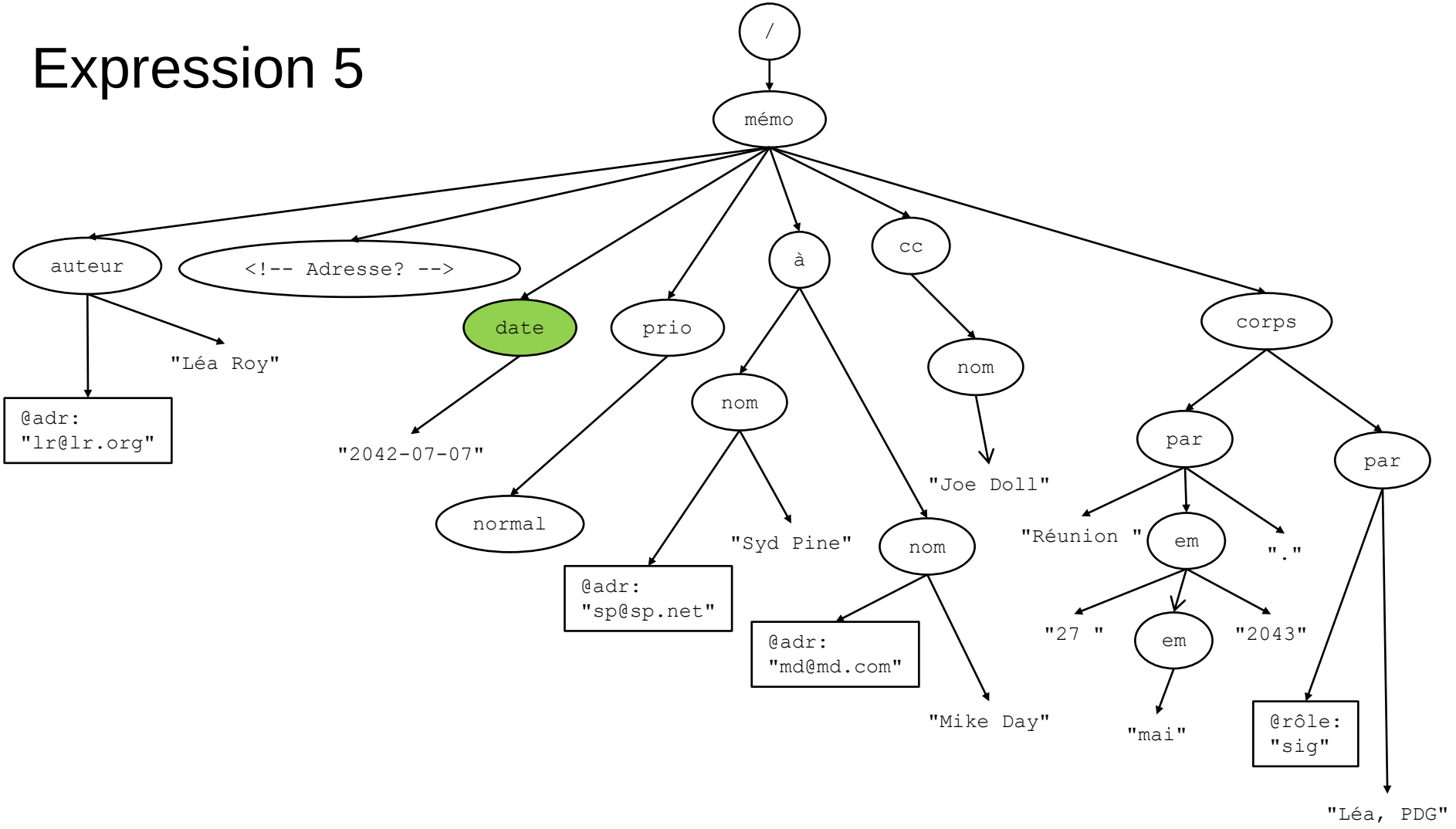
Expression: `//@adr`

Expression 5



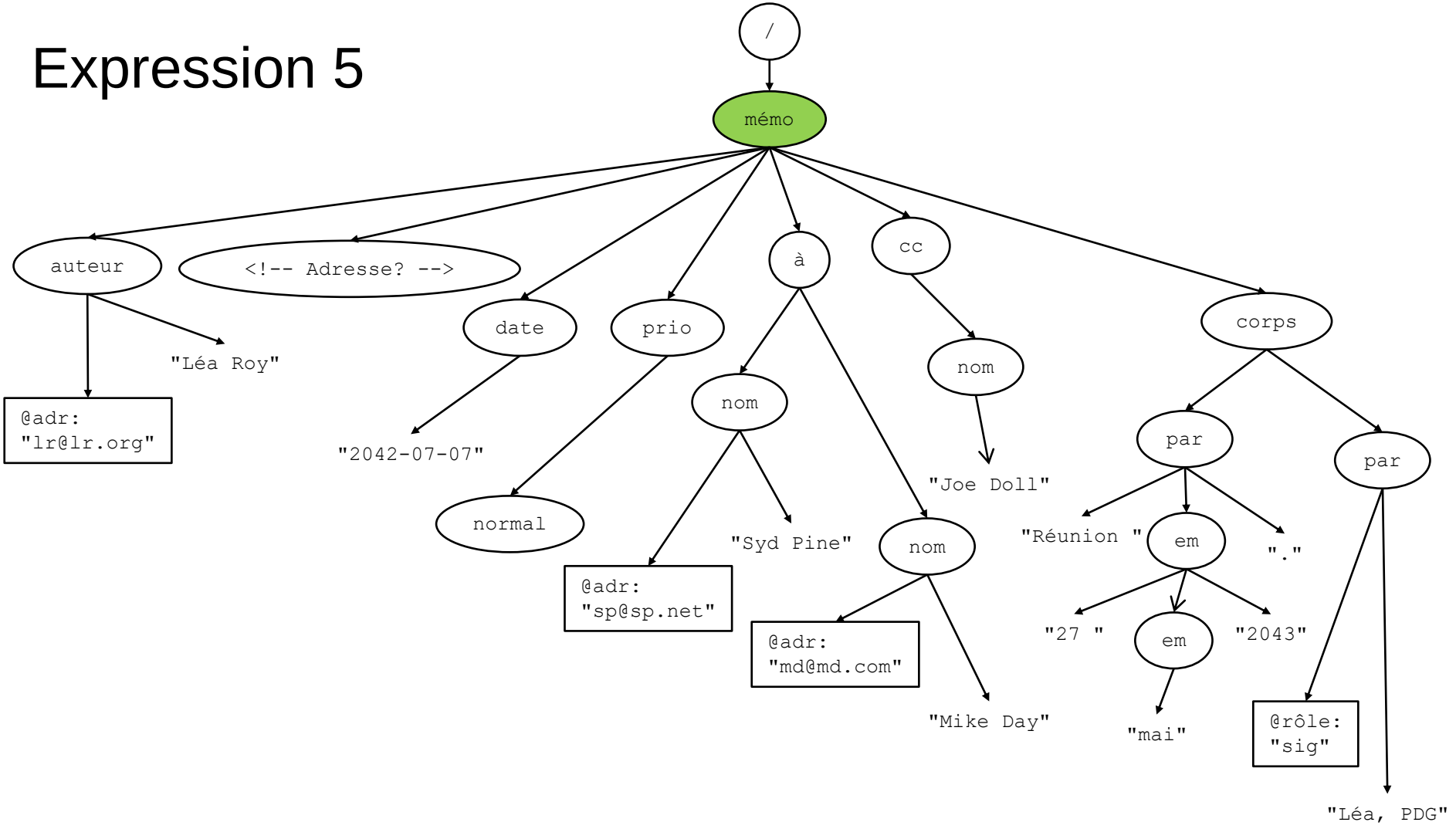
Expression: `//date/../*`

Expression 5



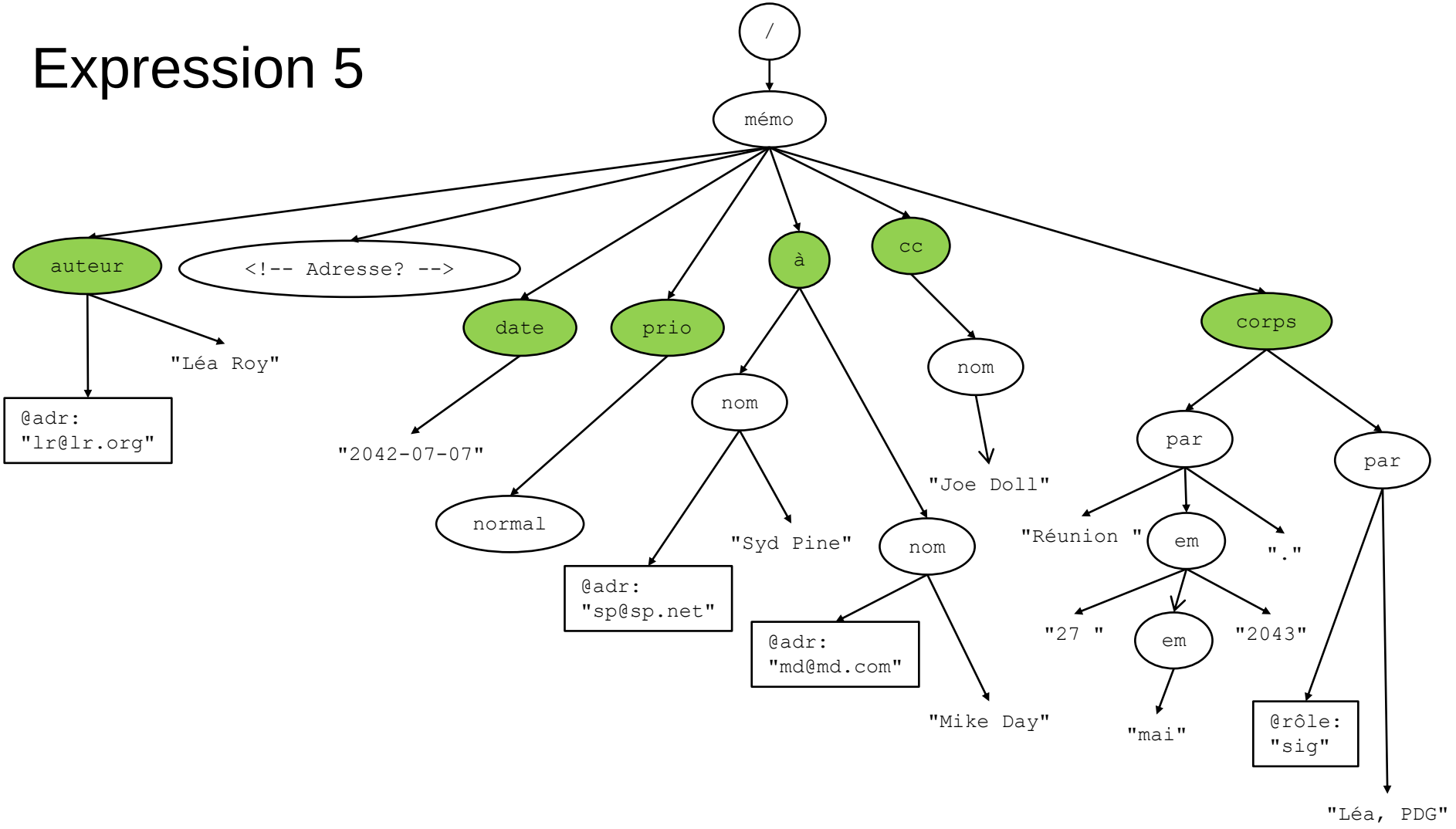
Expression: `//date/../*`

Expression 5



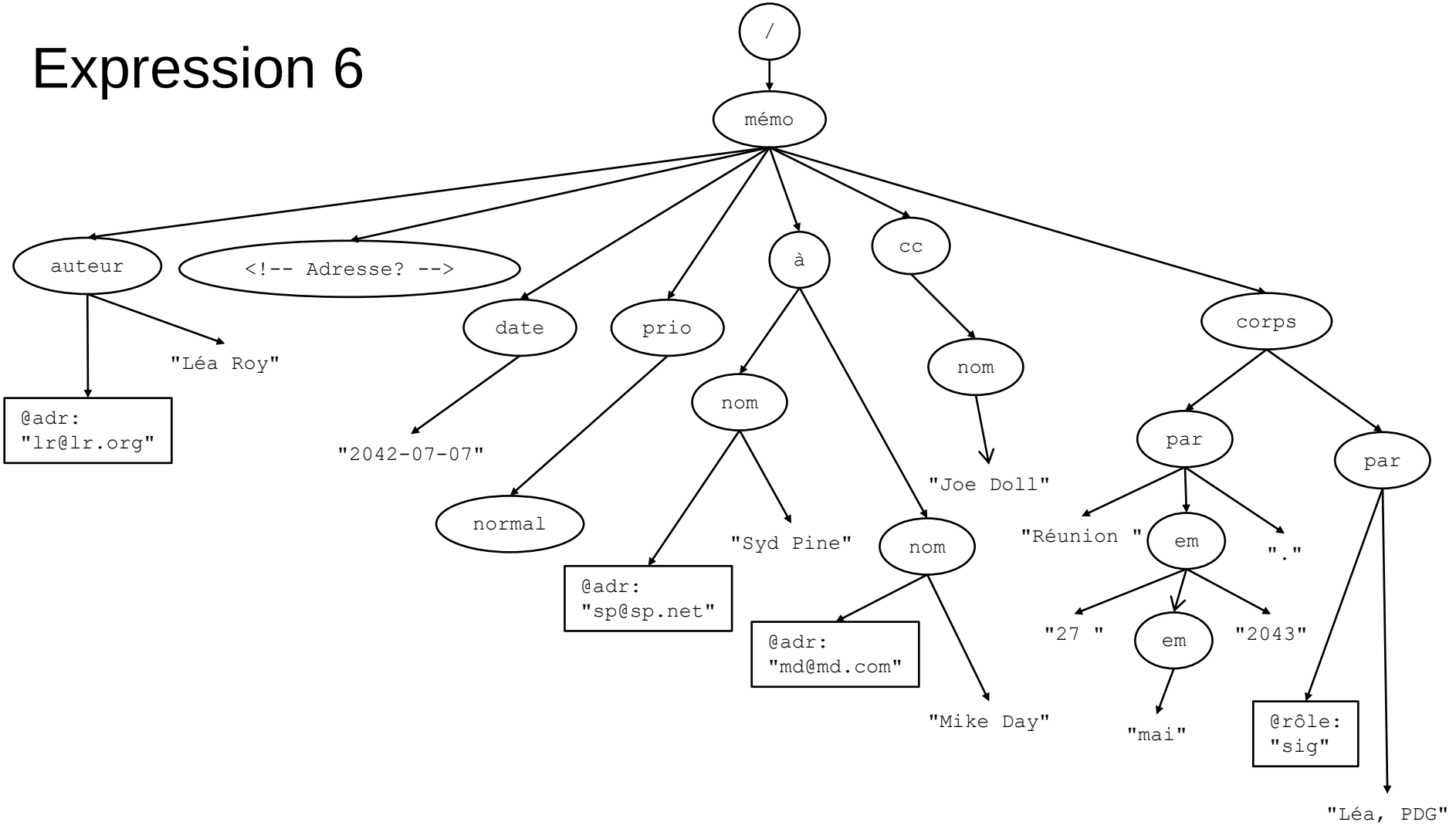
Expression: `//date/../*`

Expression 5



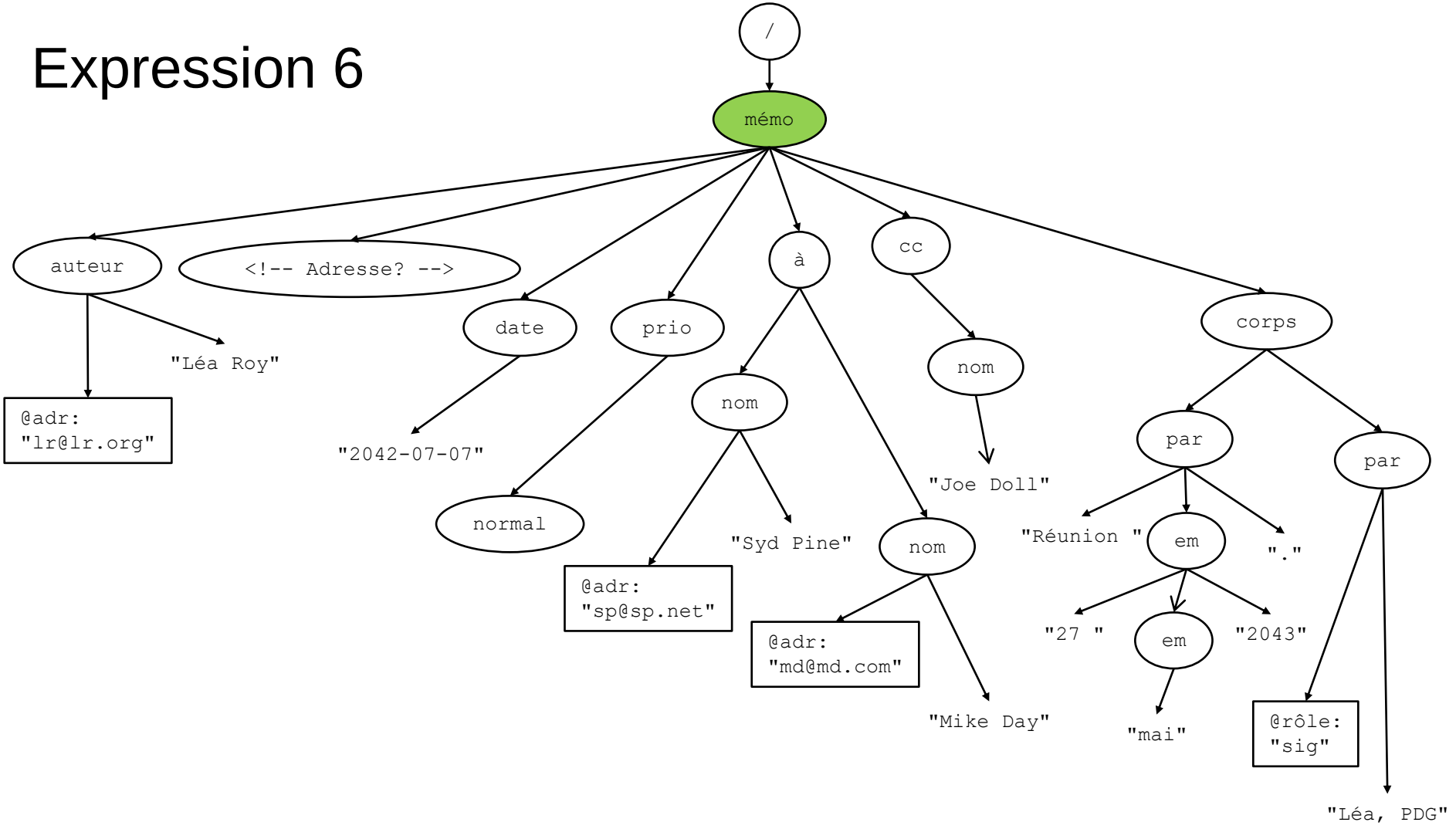
Expression: `//date/../*`

Expression 6



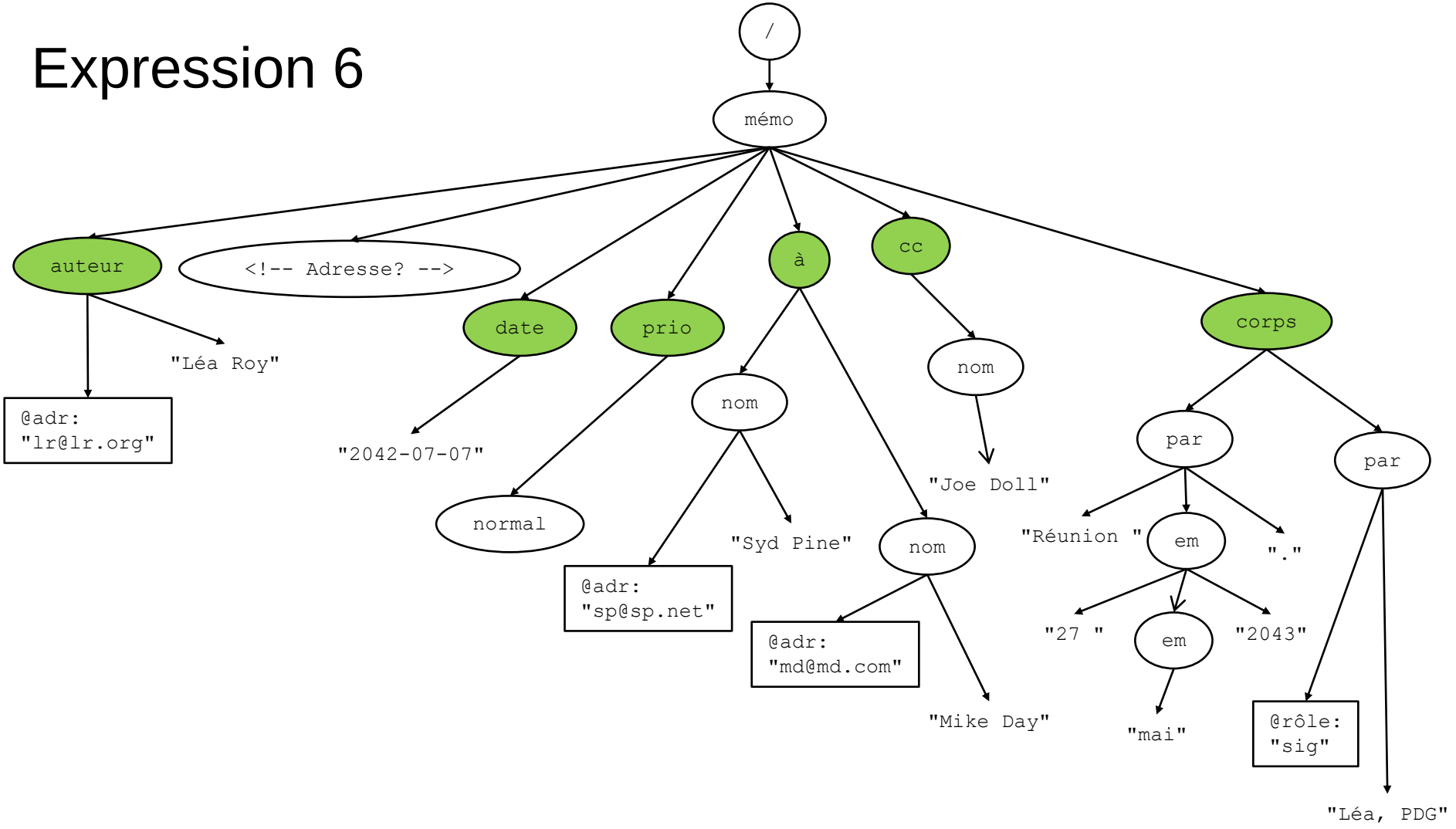
Expression: `/*/*[.//@adr]`

Expression 6



Expression: `/*/*[.//@adr]`

Expression 6

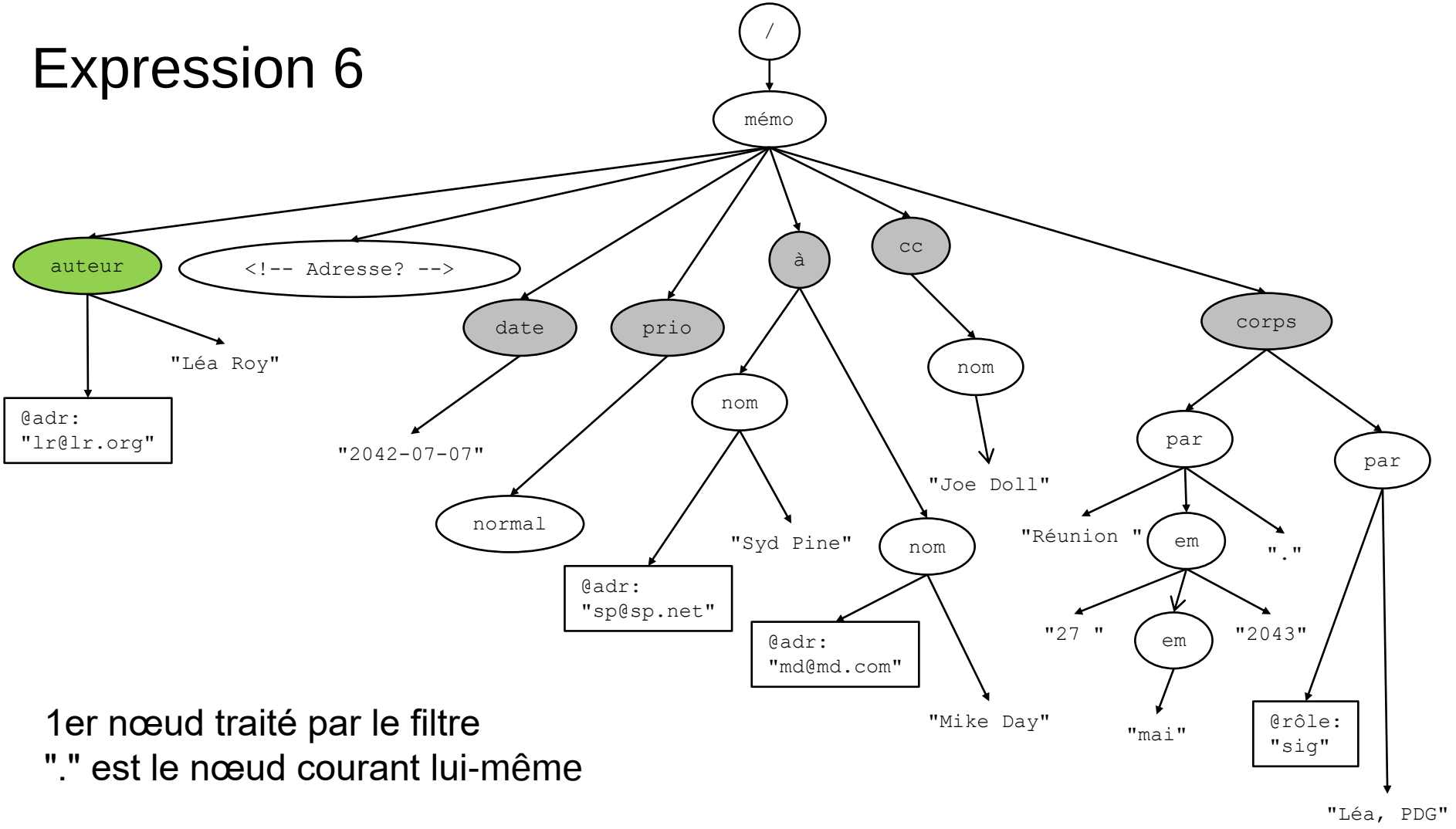


Expression: `/*/*[.//@adr]`

Fonctionnement du filtre

- Un filtre est une expression XPath entre [ ]
- Ici, le filtre est un *chemin XPath* ("Cas 1")
- Chaque nœud *n* soumis au filtre est tour à tour pris comme nœud courant :
  - Le filtre (chemin relatif) est exécuté avec *n* comme nœud courant
  - Si le résultat est *vide* (ne contient aucun nœud), *n* est *éliminé* par le filtre; sinon, il est *gardé*

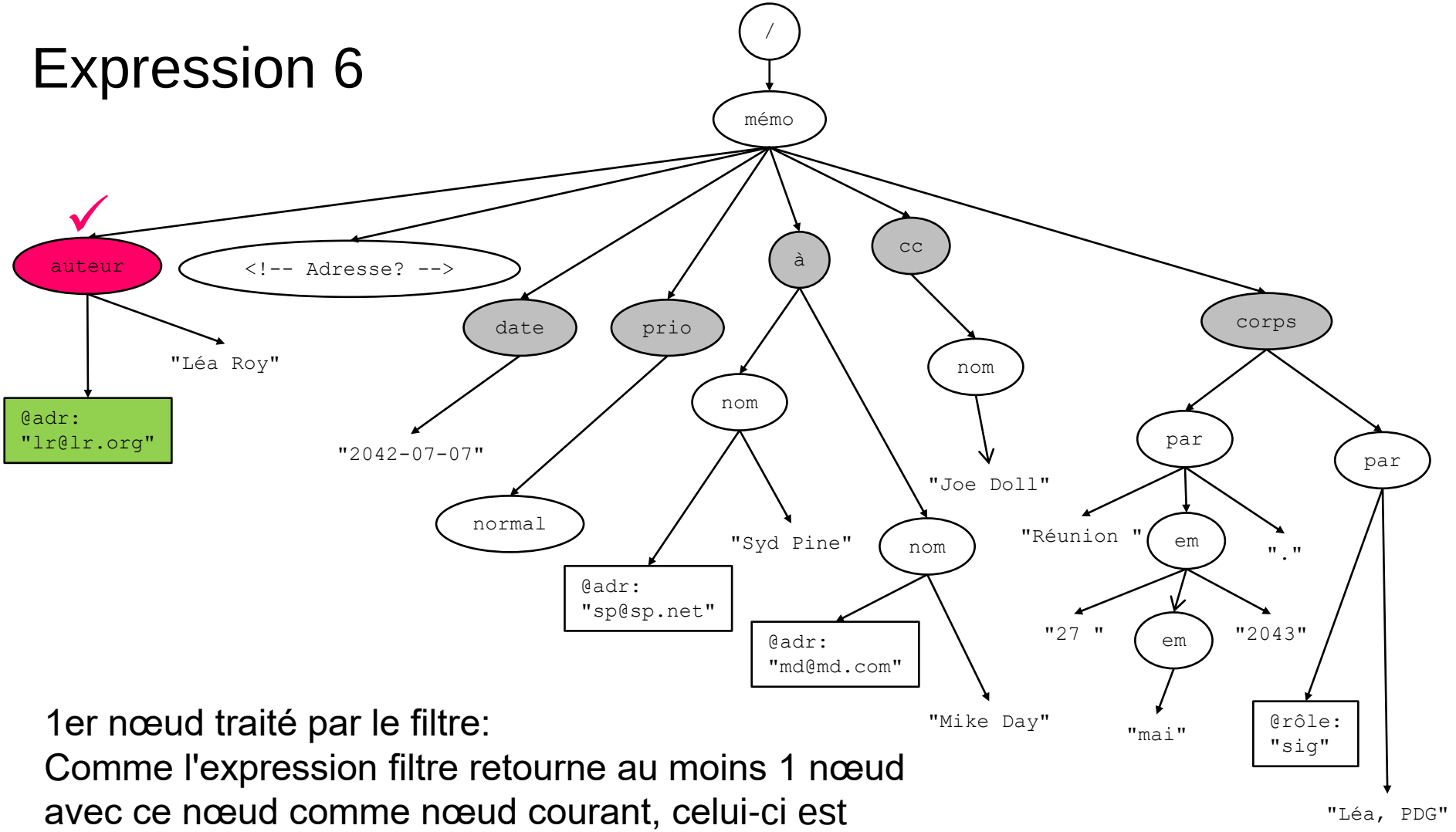
Expression 6



1er nœud traité par le filtre
"." est le nœud courant lui-même

Expression: `/*/*[.//@adr]`

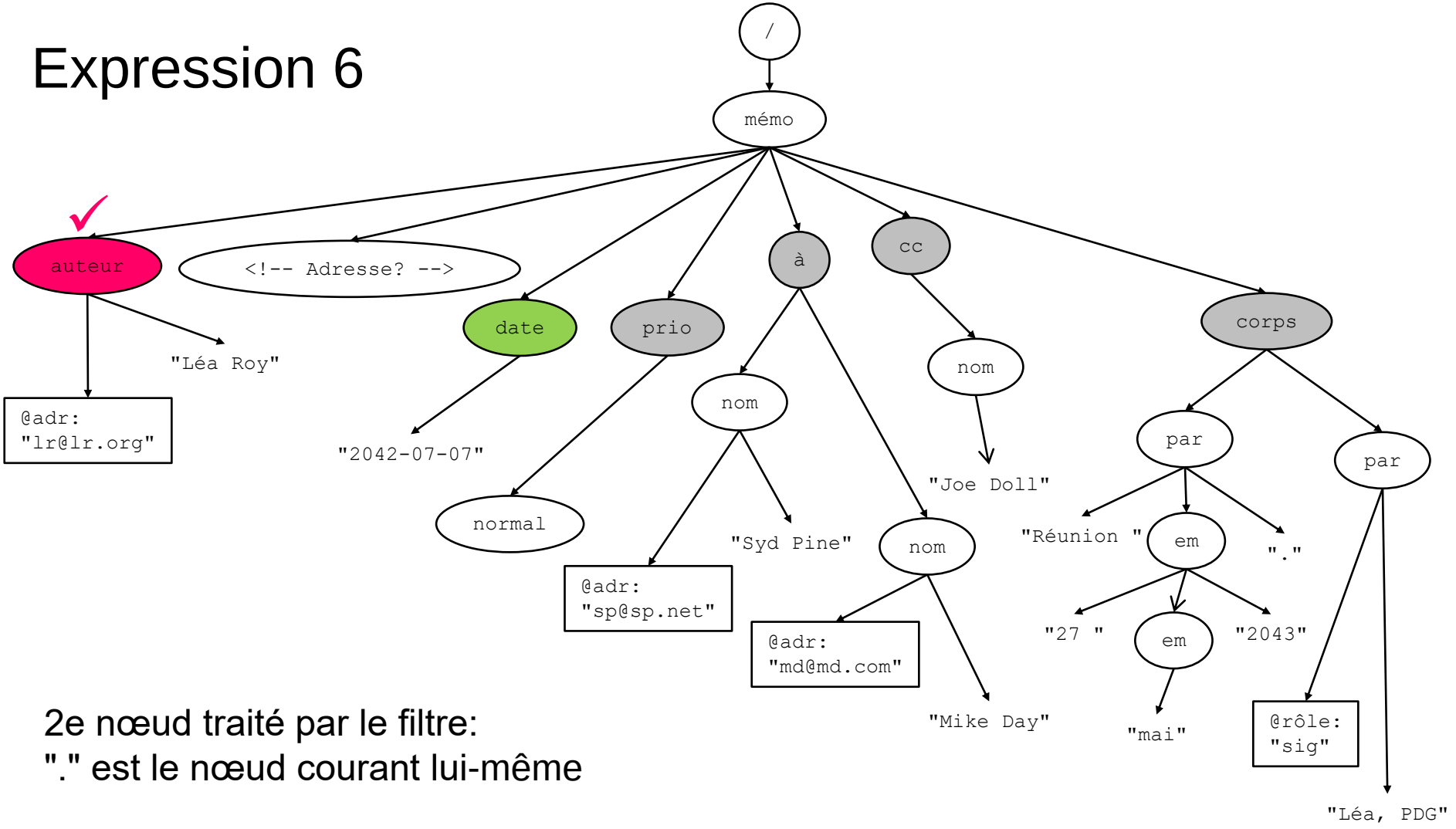
Expression 6



1er nœud traité par le filtre:
Comme l'expression filtre retourne au moins 1 nœud avec ce nœud comme nœud courant, celui-ci est gardé par le filtre

Expression: `/*/*[.//@adr]`

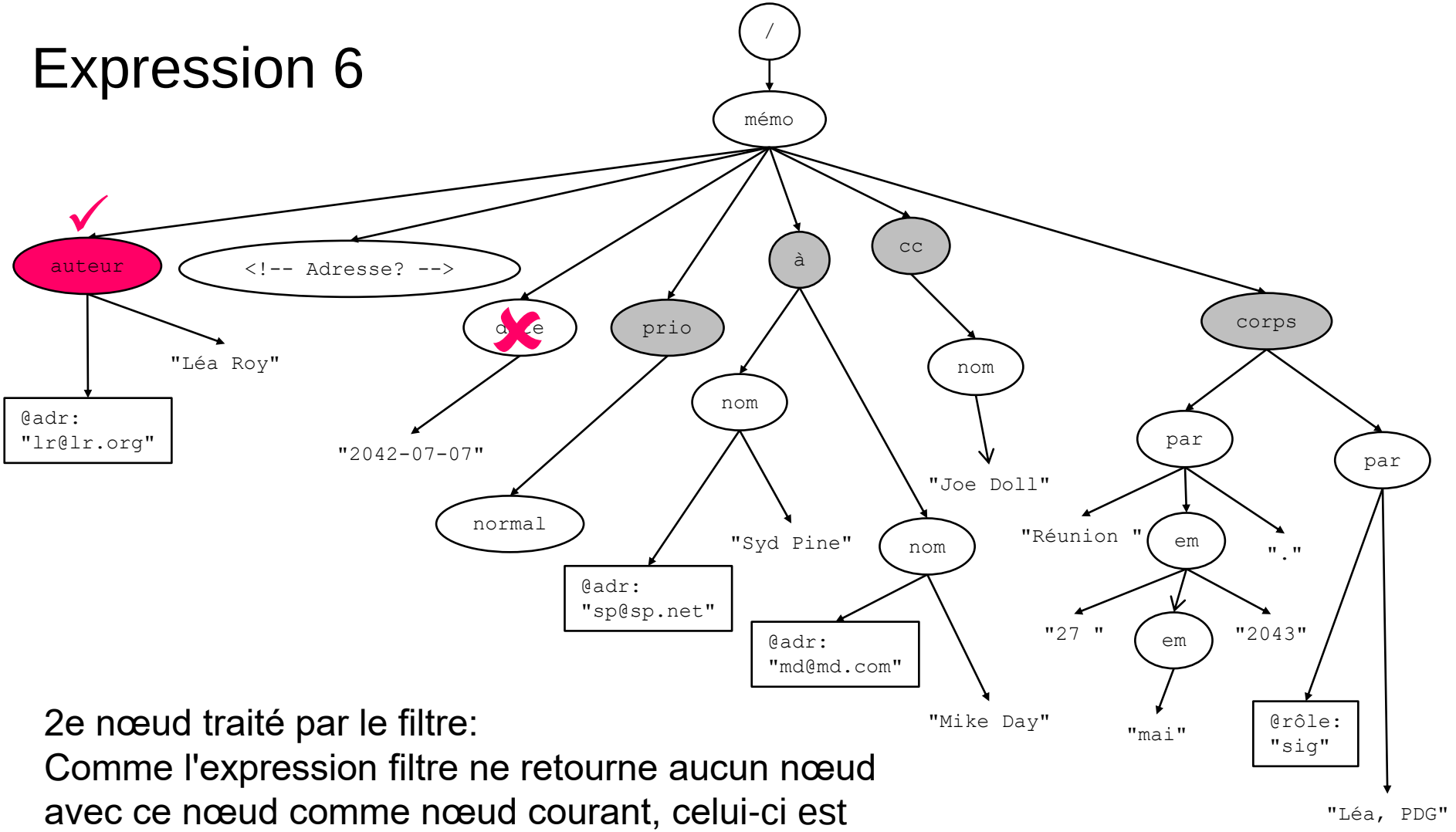
Expression 6



2e nœud traité par le filtre:
"." est le nœud courant lui-même

Expression: `/*/*[.//@adr]`

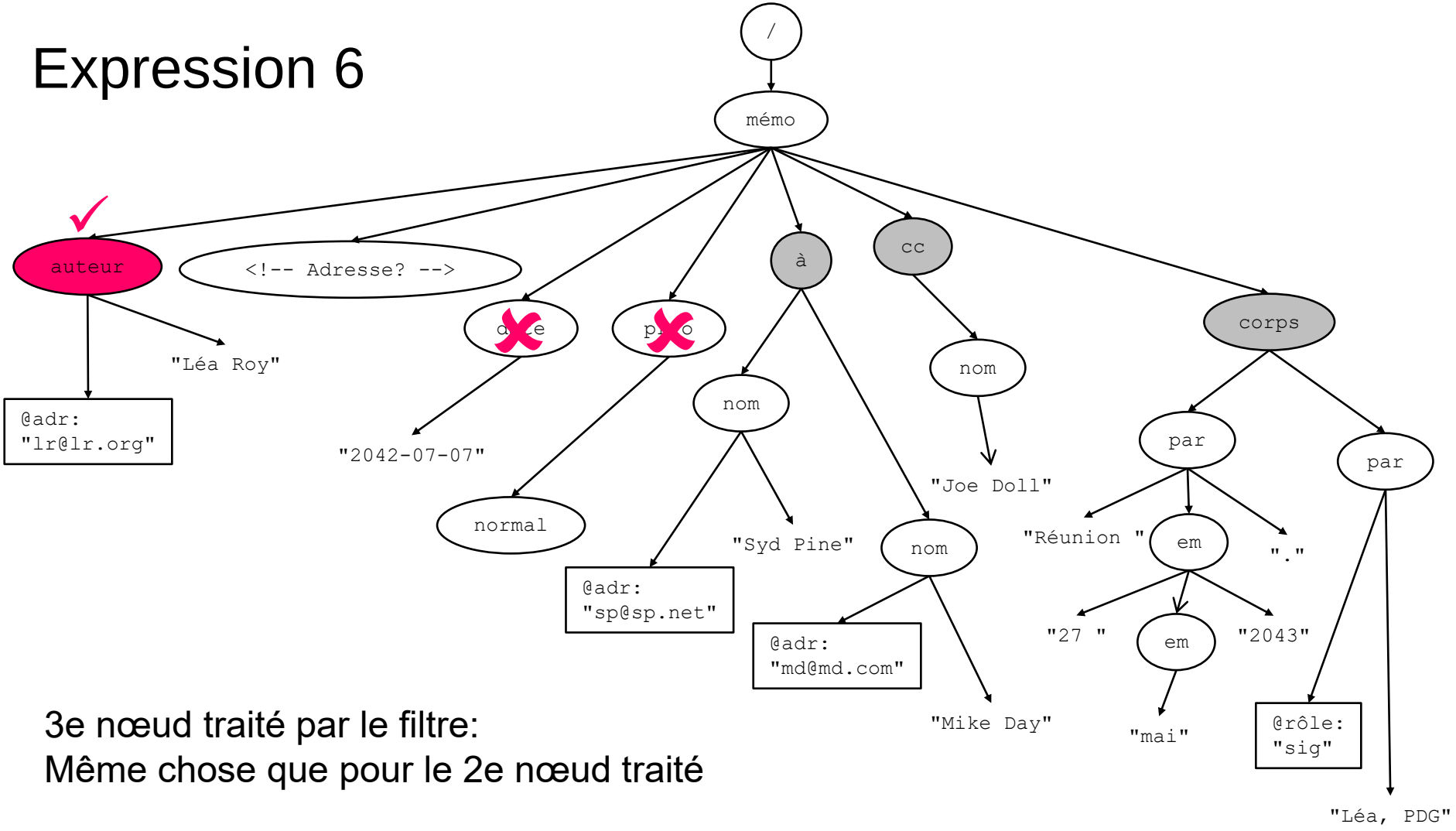
Expression 6



2e nœud traité par le filtre:
Comme l'expression filtre ne retourne aucun nœud avec ce nœud comme nœud courant, celui-ci est éliminé par le filtre

Expression: `/*/*[.//@adr]`

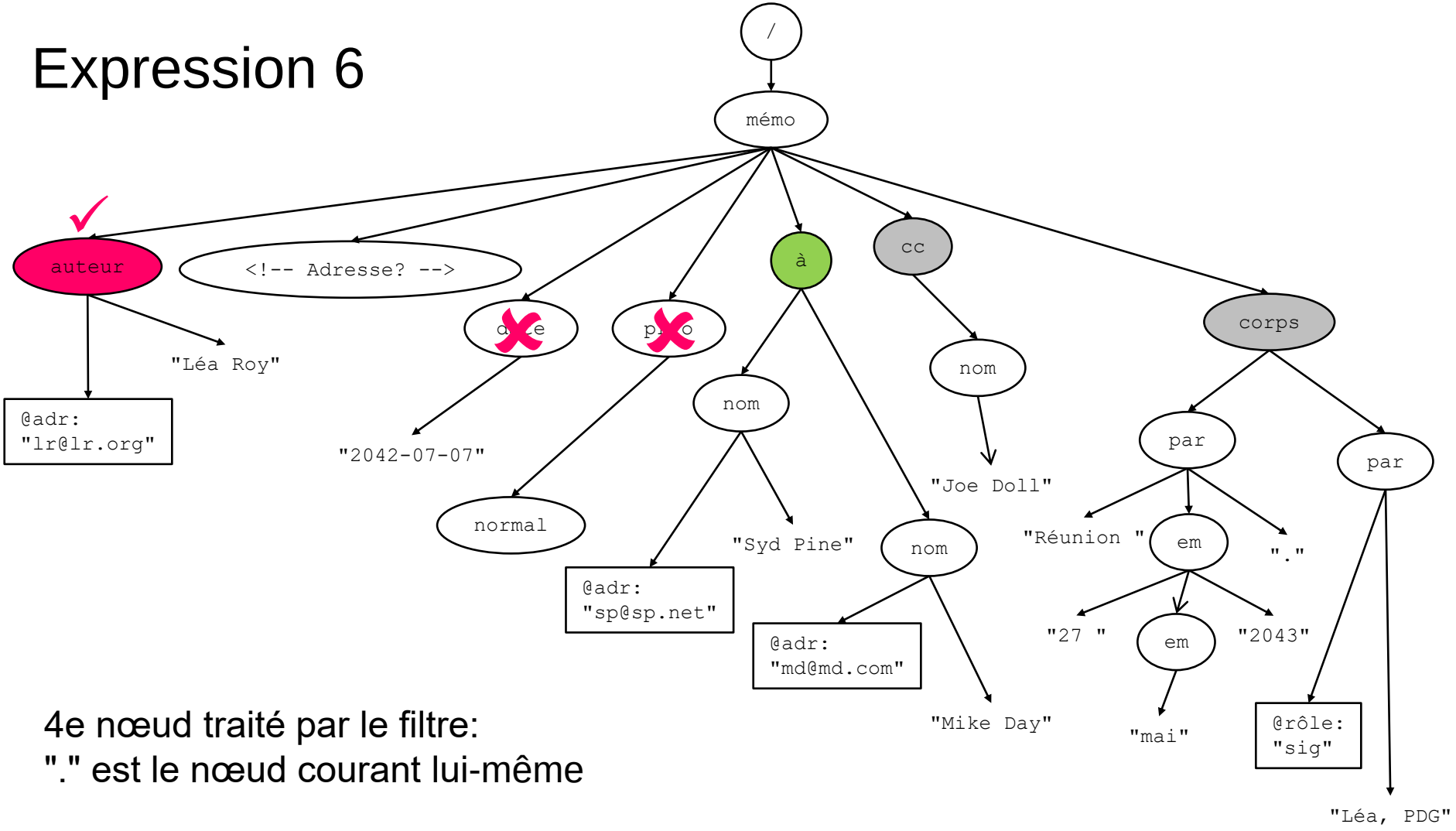
Expression 6



3e nœud traité par le filtre:
Même chose que pour le 2e nœud traité

Expression: `/*/*[.//@adr]`

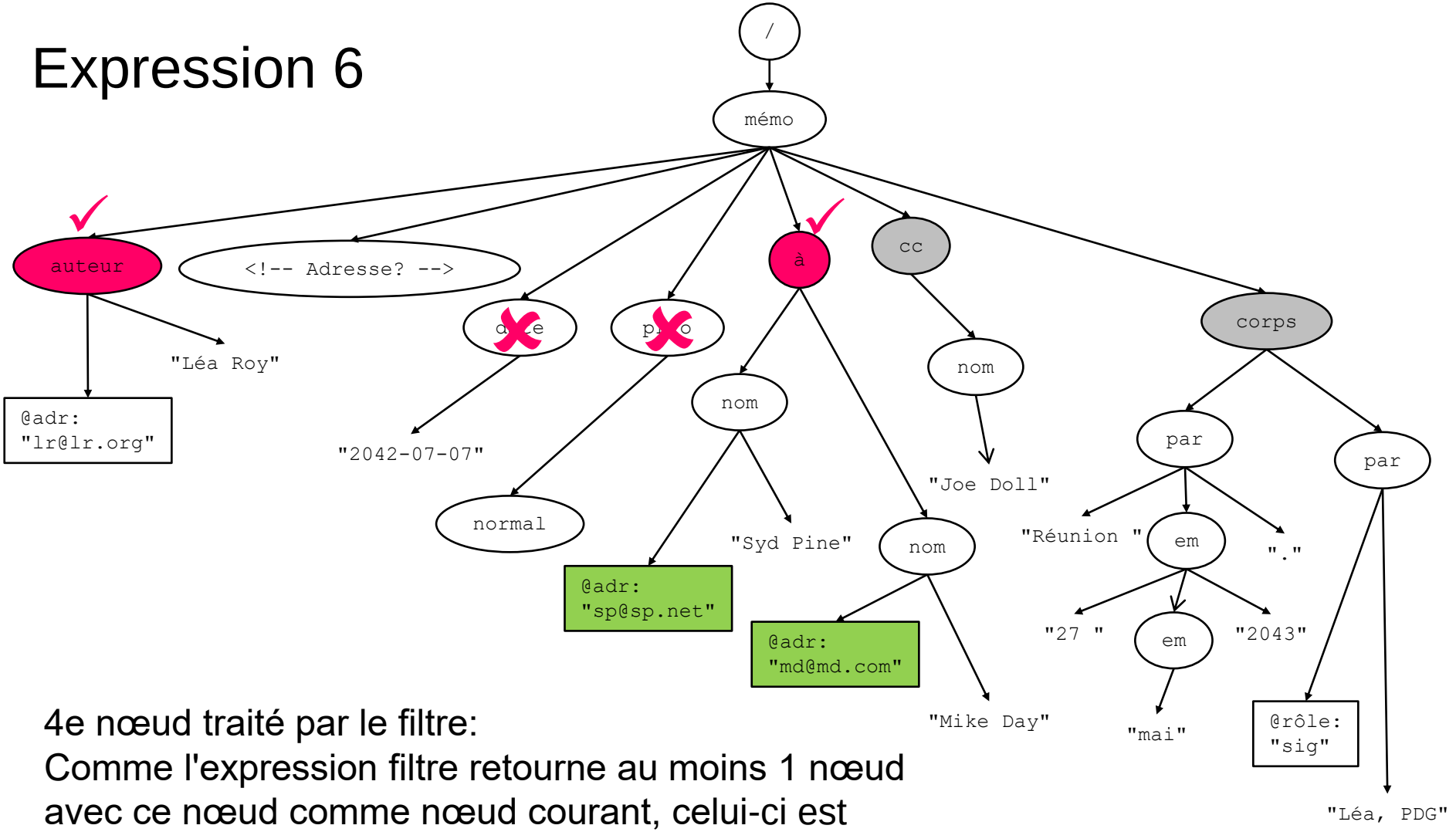
Expression 6



4e nœud traité par le filtre:
"." est le nœud courant lui-même

Expression: `/*/*[.//@adr]`

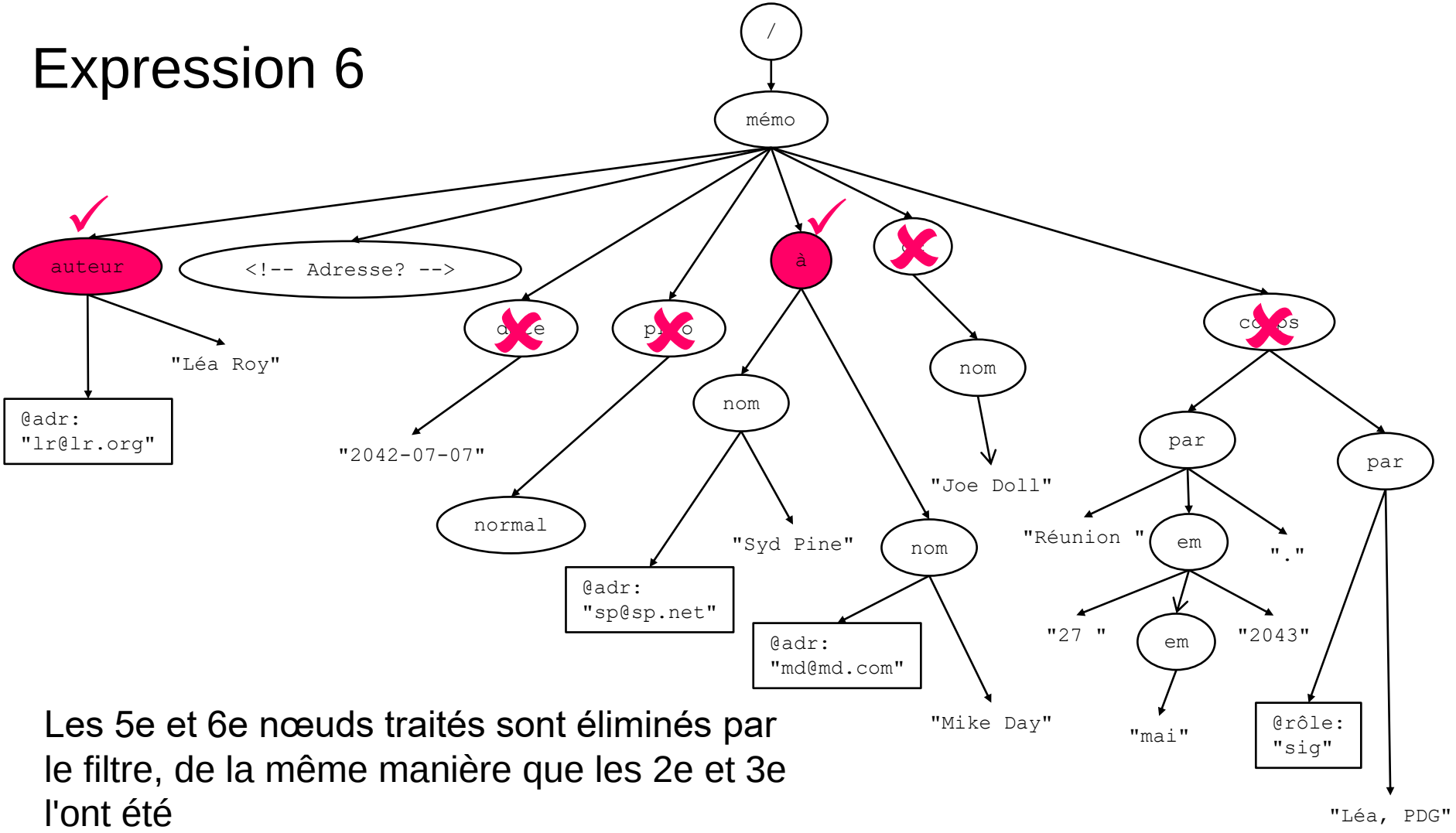
Expression 6



4e nœud traité par le filtre:
Comme l'expression filtre retourne au moins 1 nœud avec ce nœud comme nœud courant, celui-ci est gardé par le filtre

Expression: `/*/*[.//@adr]`

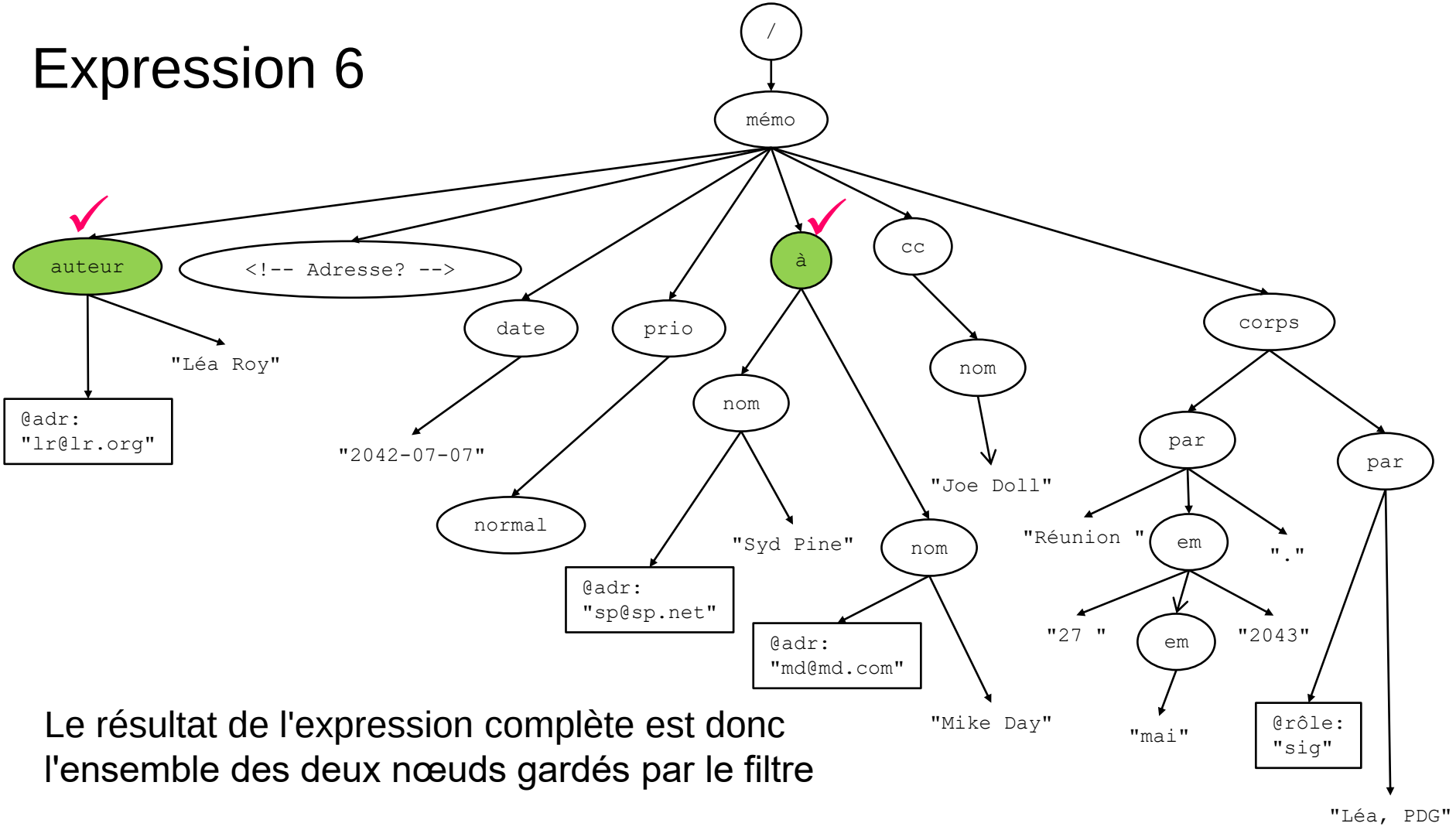
Expression 6



Les 5e et 6e nœuds traités sont éliminés par le filtre, de la même manière que les 2e et 3e l'ont été

Expression: `/*/*[.//@adr]`

Expression 6



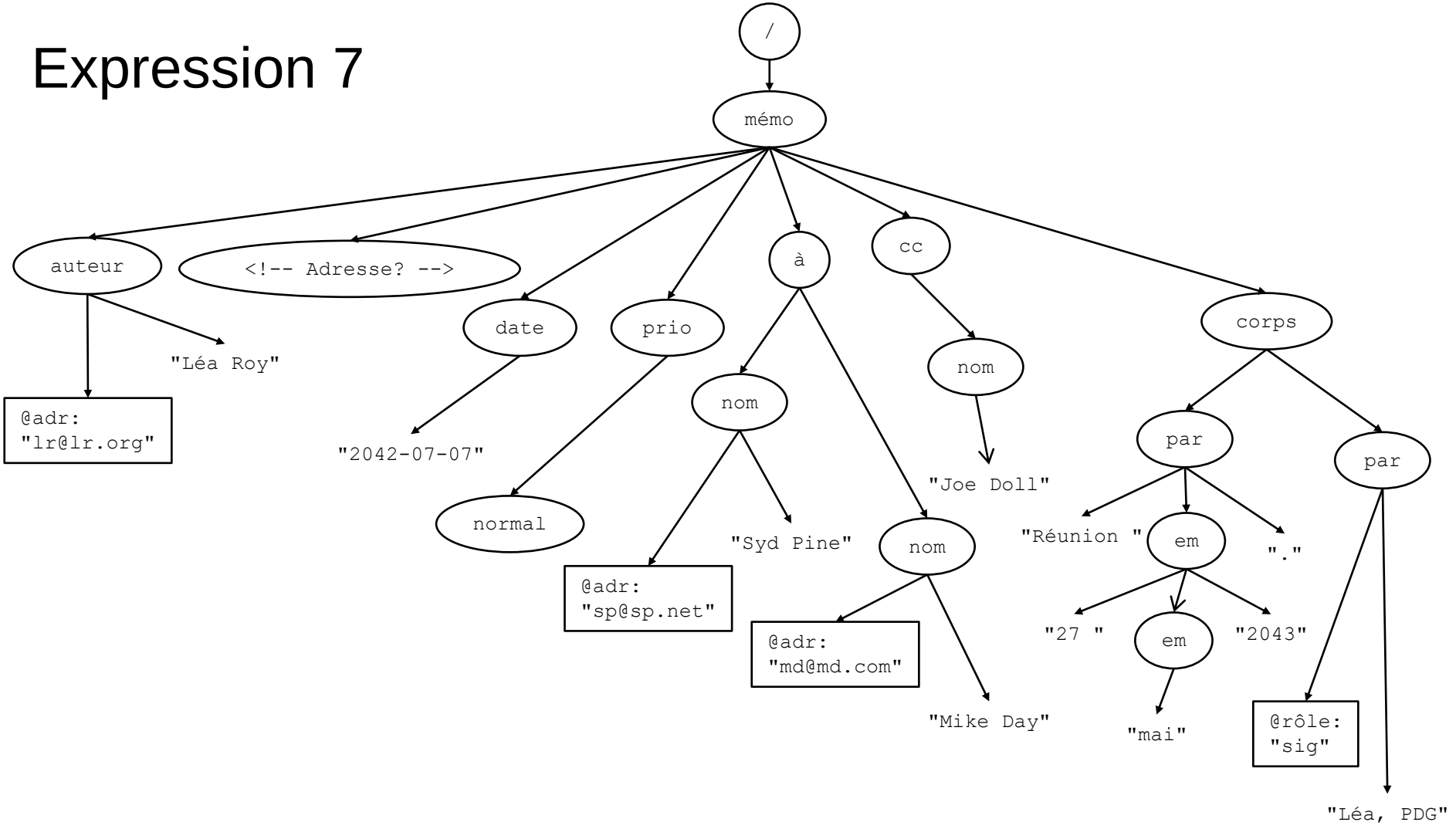
Le résultat de l'expression complète est donc l'ensemble des deux nœuds gardés par le filtre

Expression: `/*/*[.//@adr]`

Note

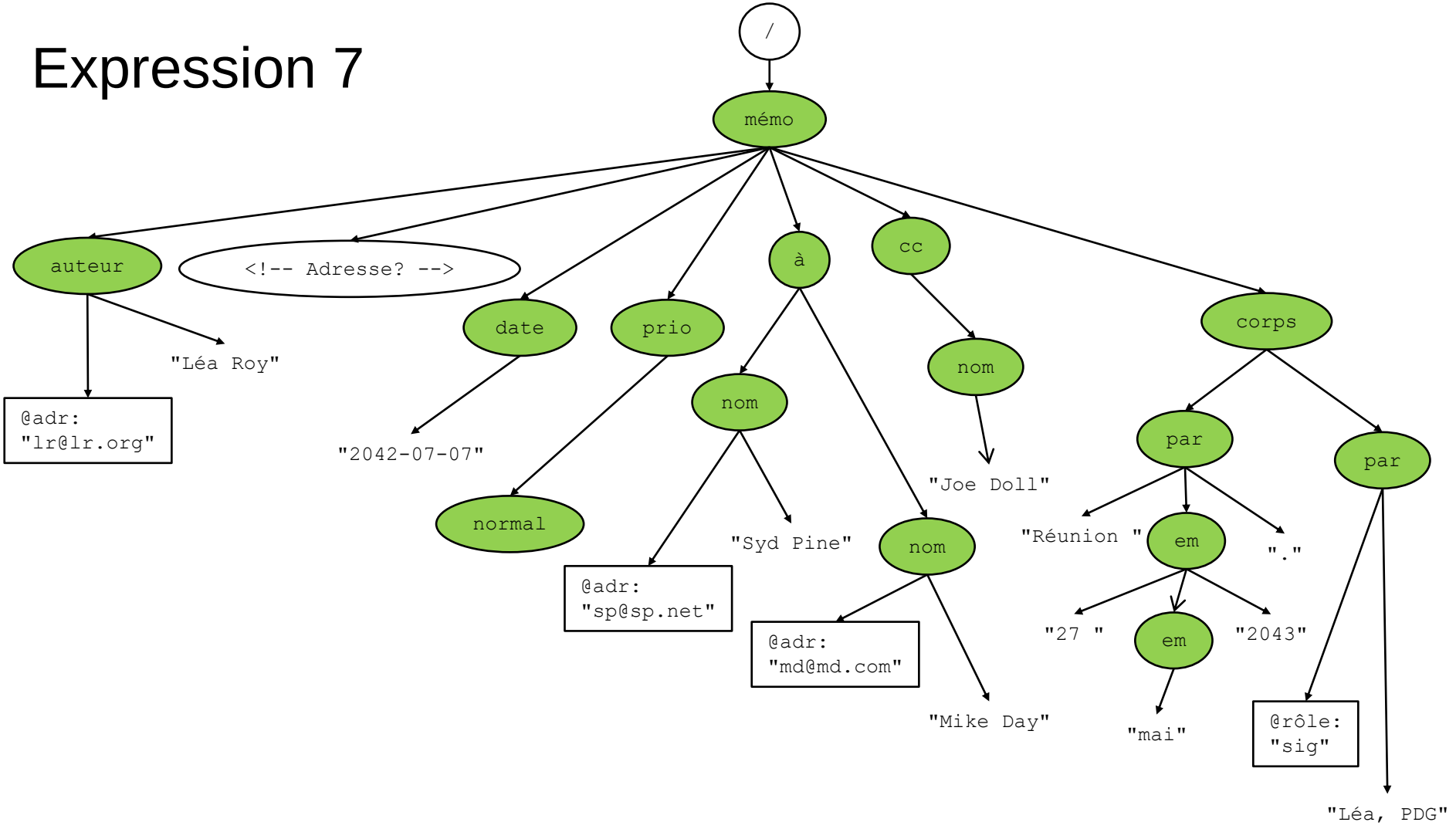
- Dans les exemples suivants, le fonctionnement des filtres n'est pas autant détaillé que dans l'exemple précédent
- Certains filtres sont des "Cas 2", i.e. des expressions XPath qui ne sont pas des chemins, mais retournent une valeur :
  - booléenne, ou
  - numérique (pour un filtre positionnel)

Expression 7



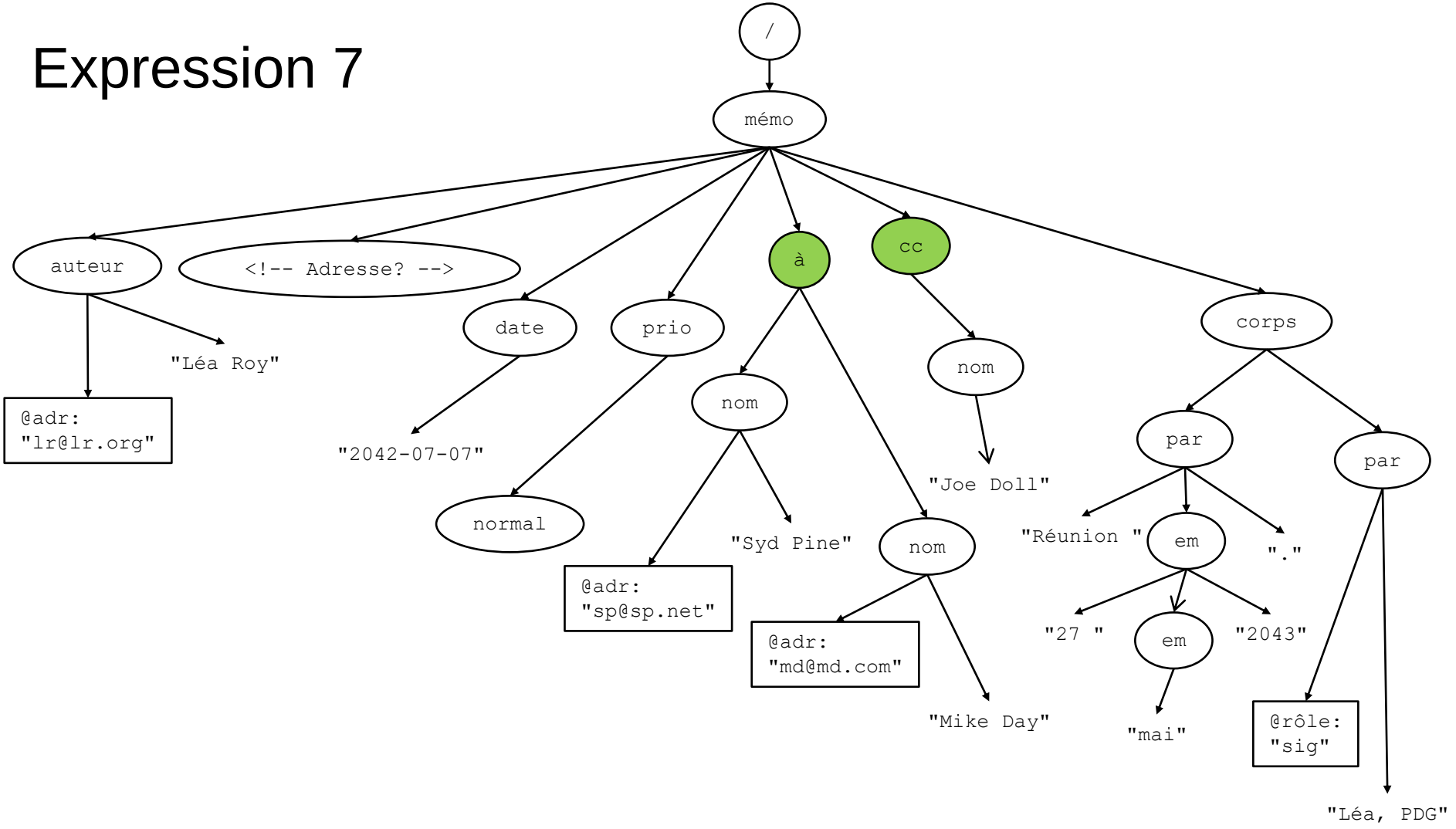
Expression: `//*[nom]`

Expression 7



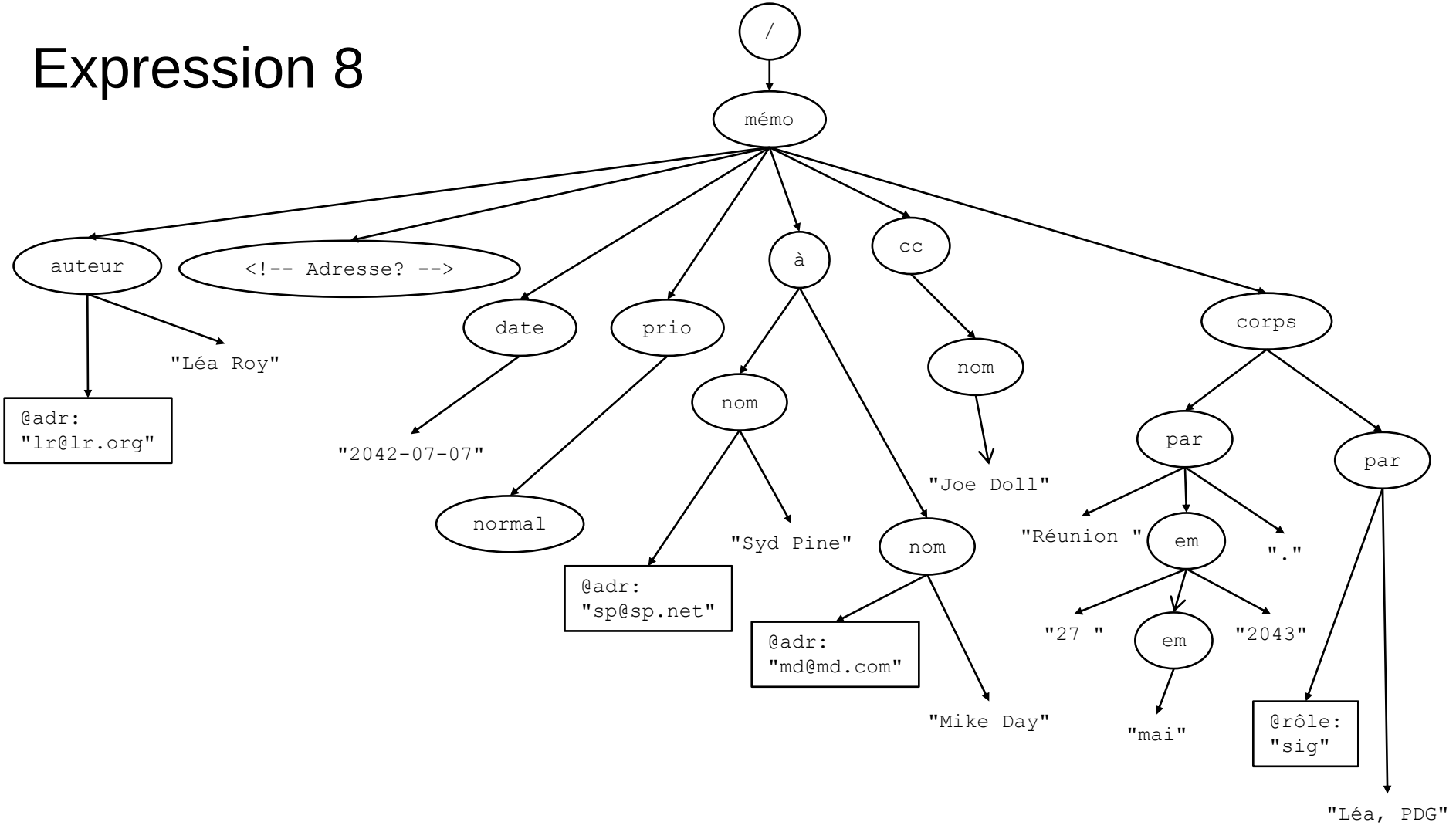
Expression: `//*[nom]`

Expression 7



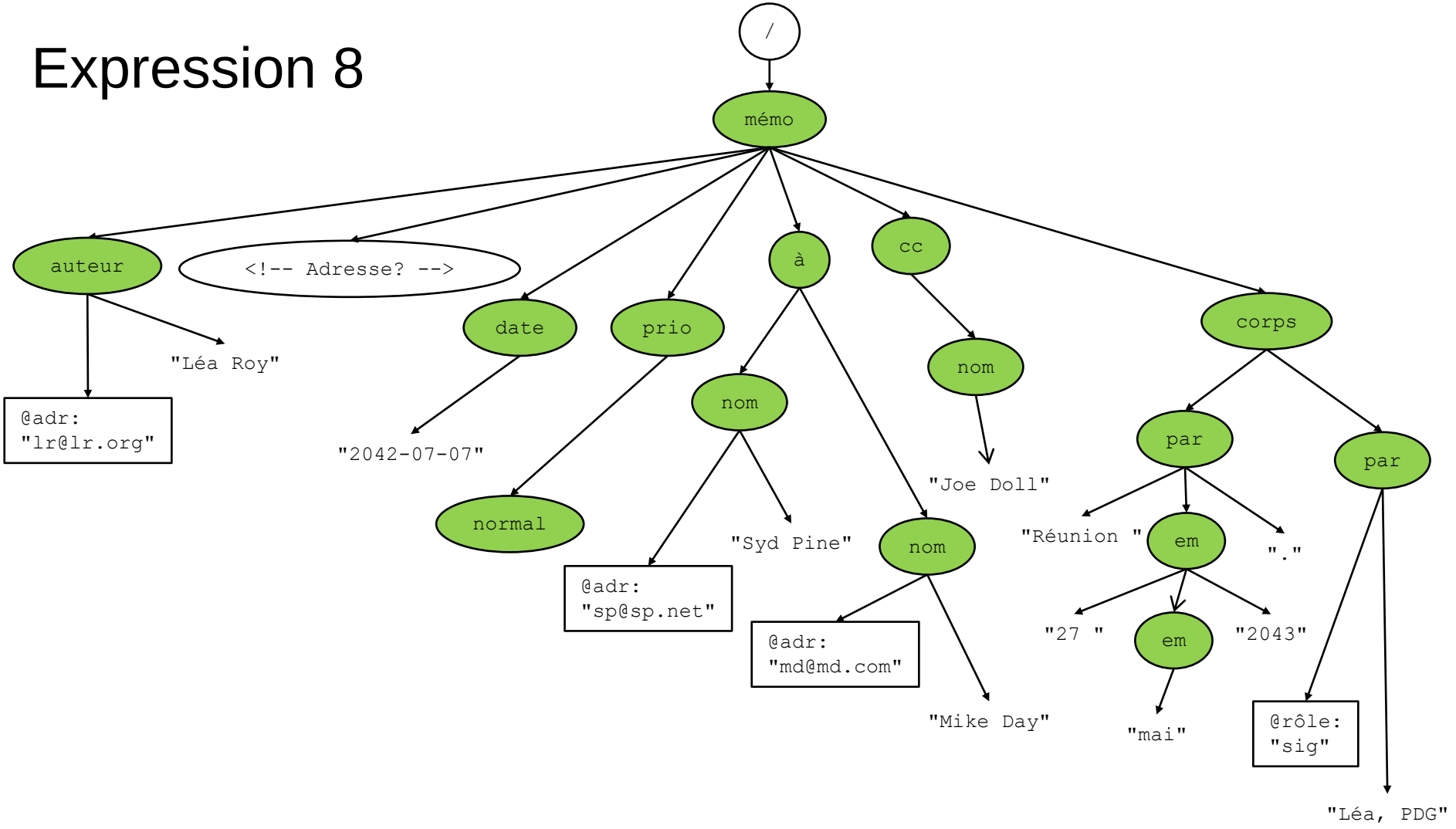
Expression: `//*[nom]`

Expression 8



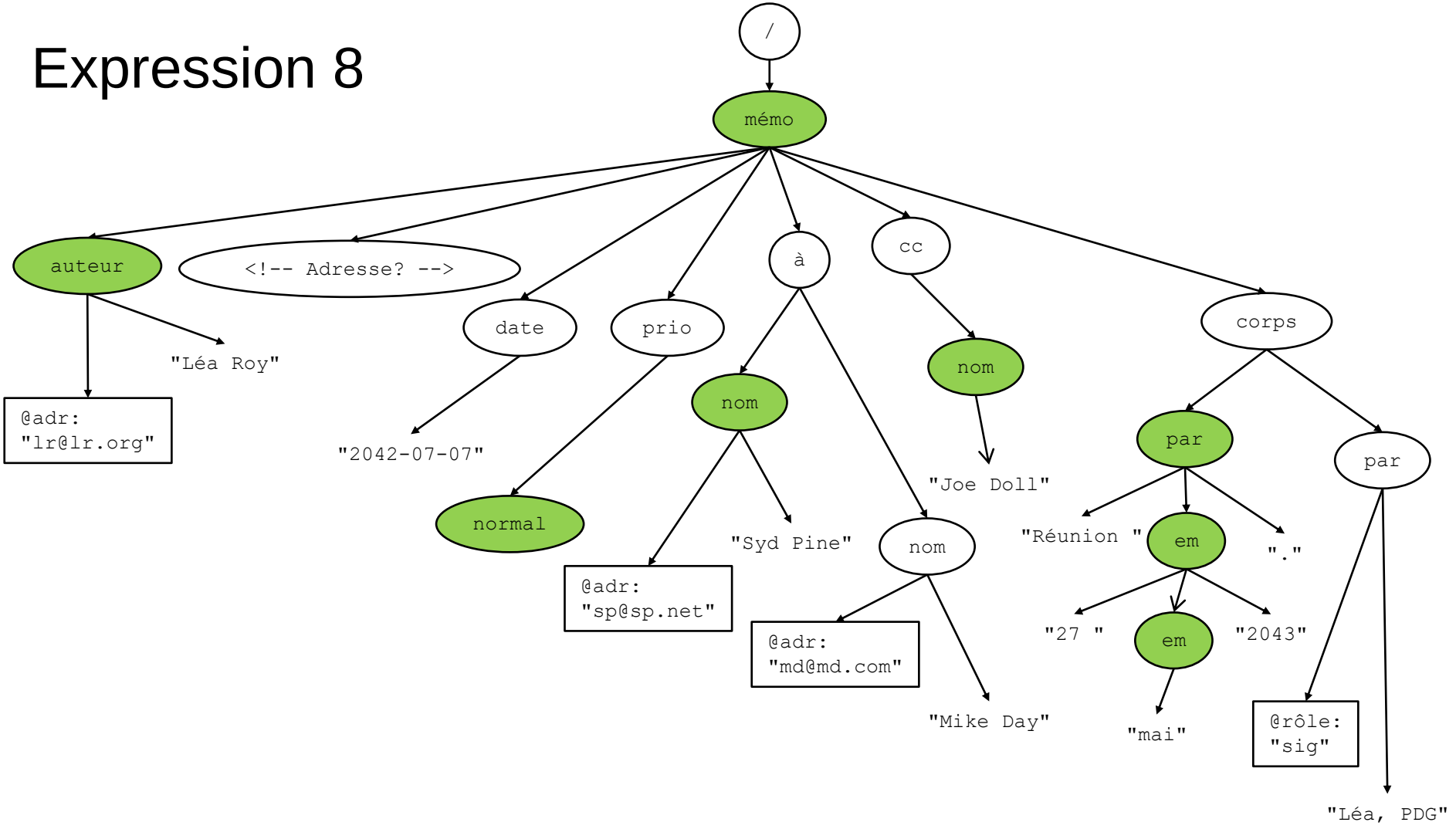
Expression: `//*[1]`

Expression 8



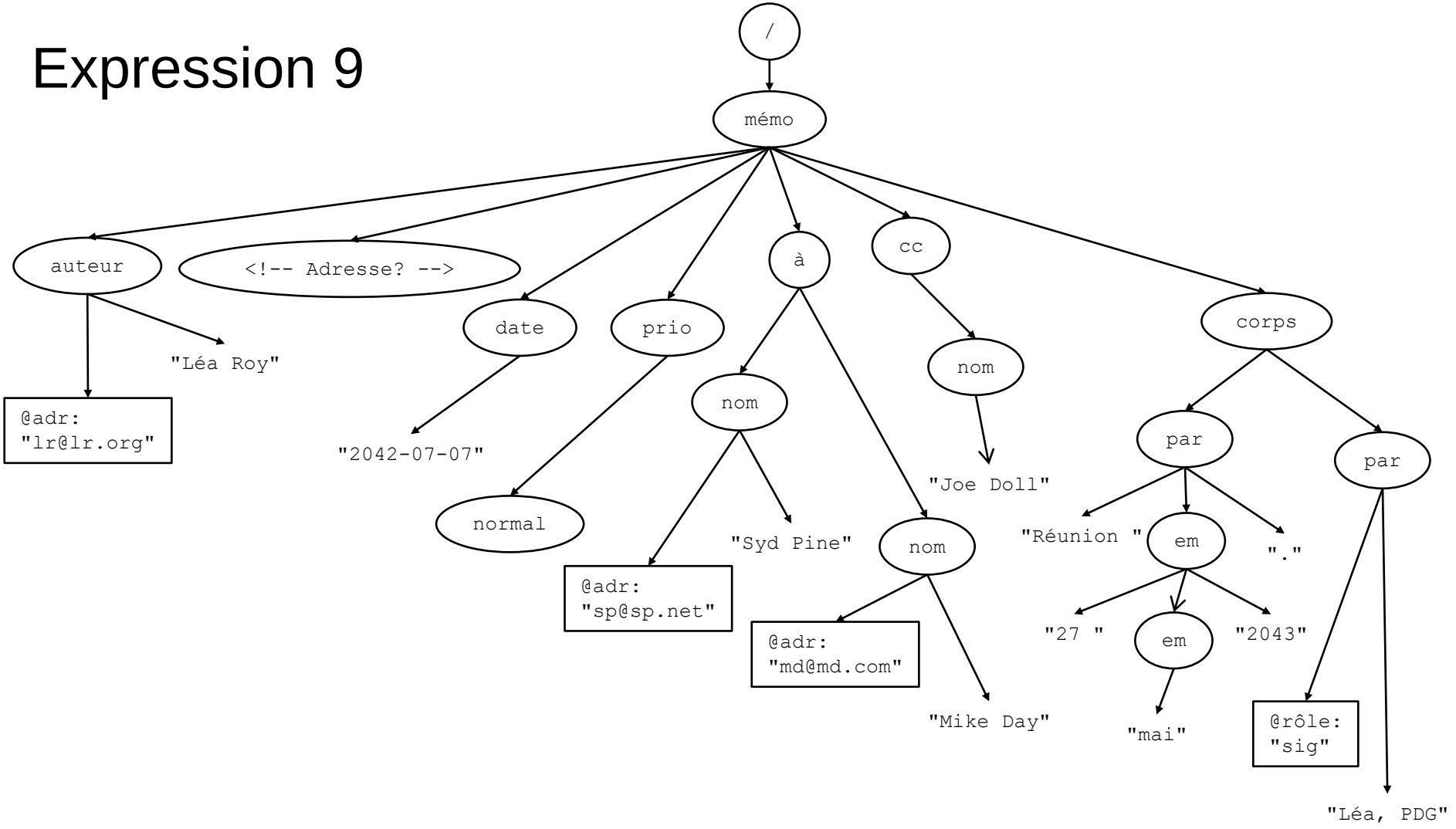
Expression: `//*[1]`

Expression 8



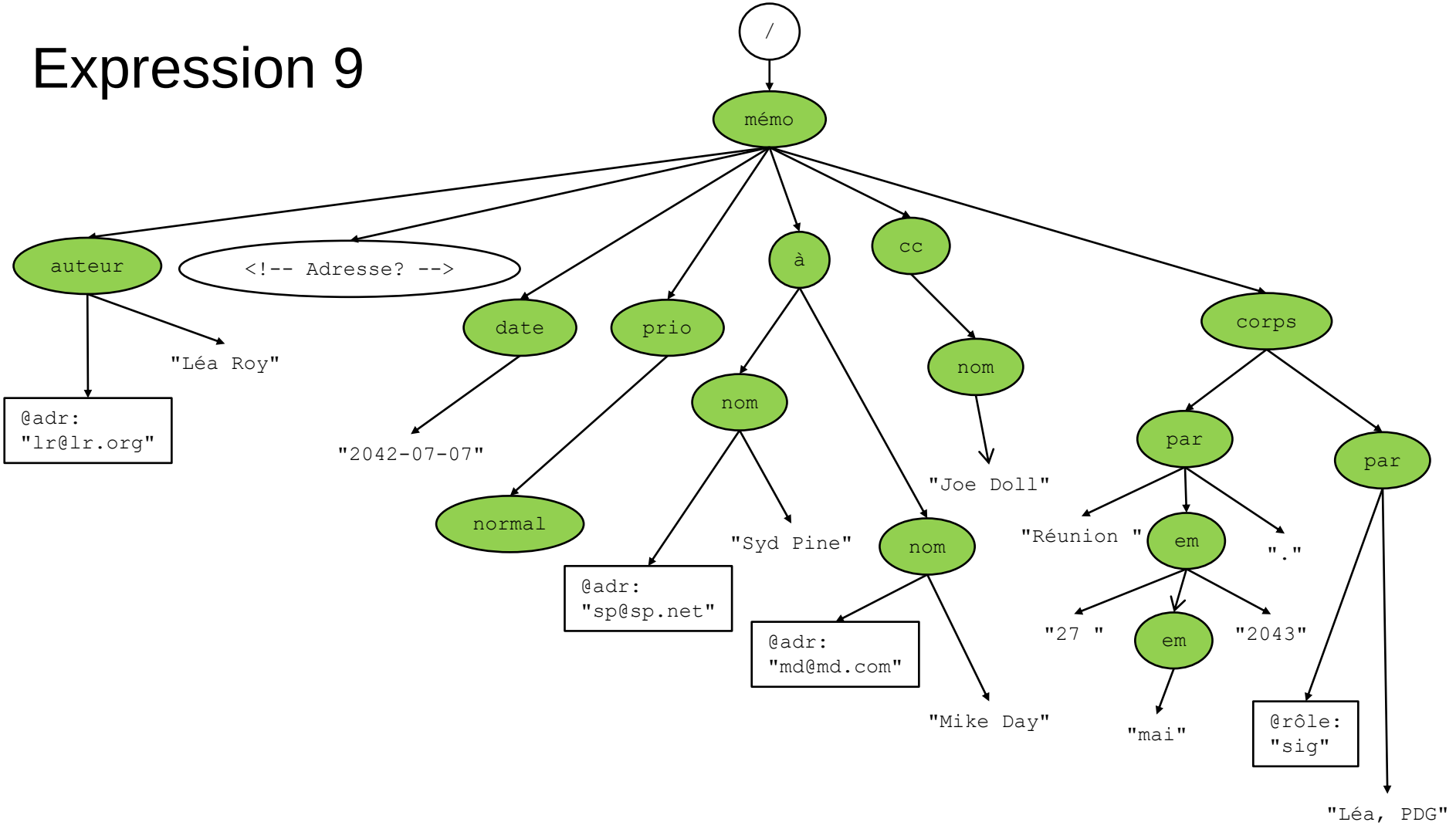
Expression: `//*[1]`

Expression 9



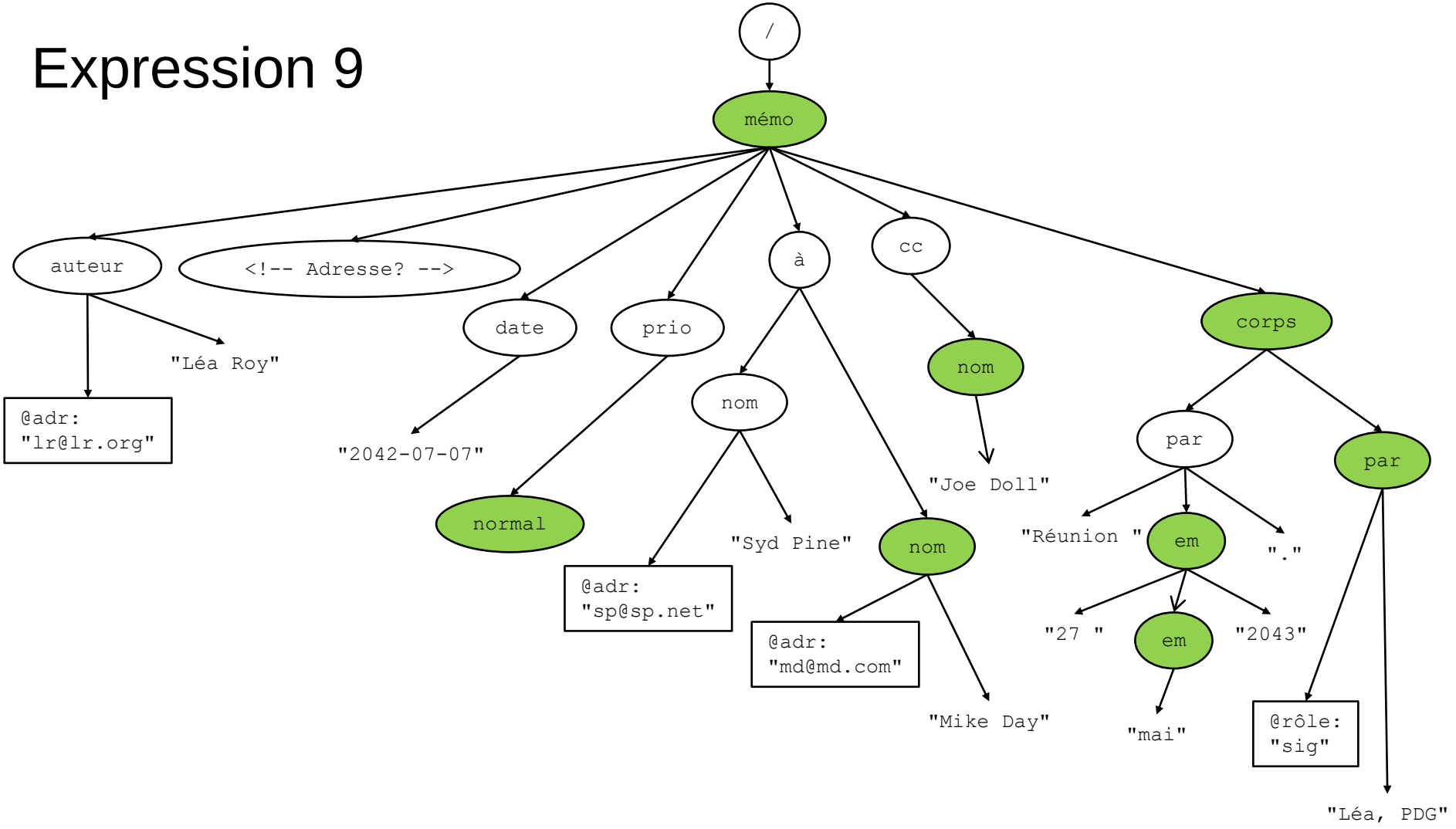
Expression: `//*[last()]`

Expression 9



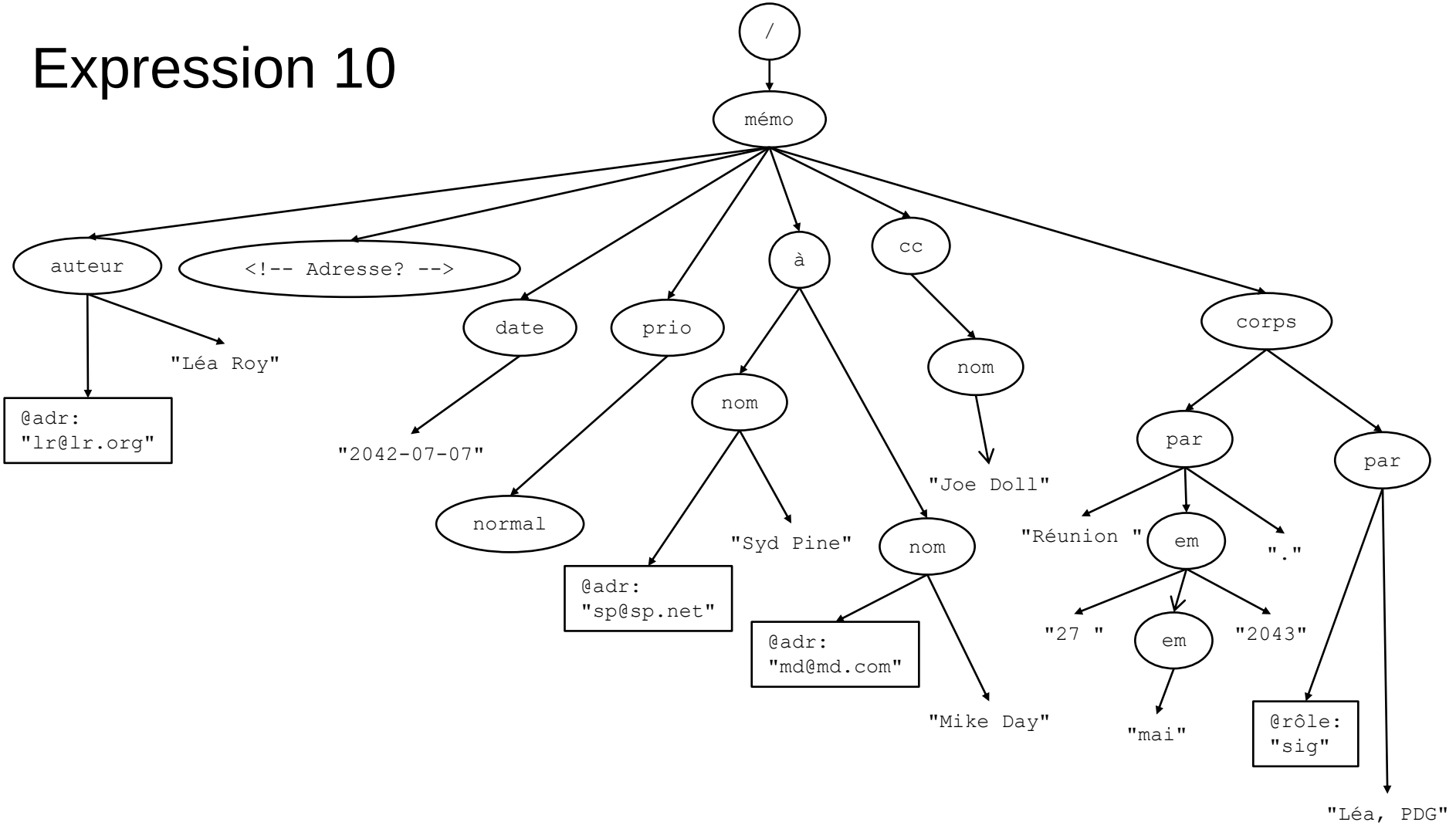
Expression: `//*[last()]`

Expression 9



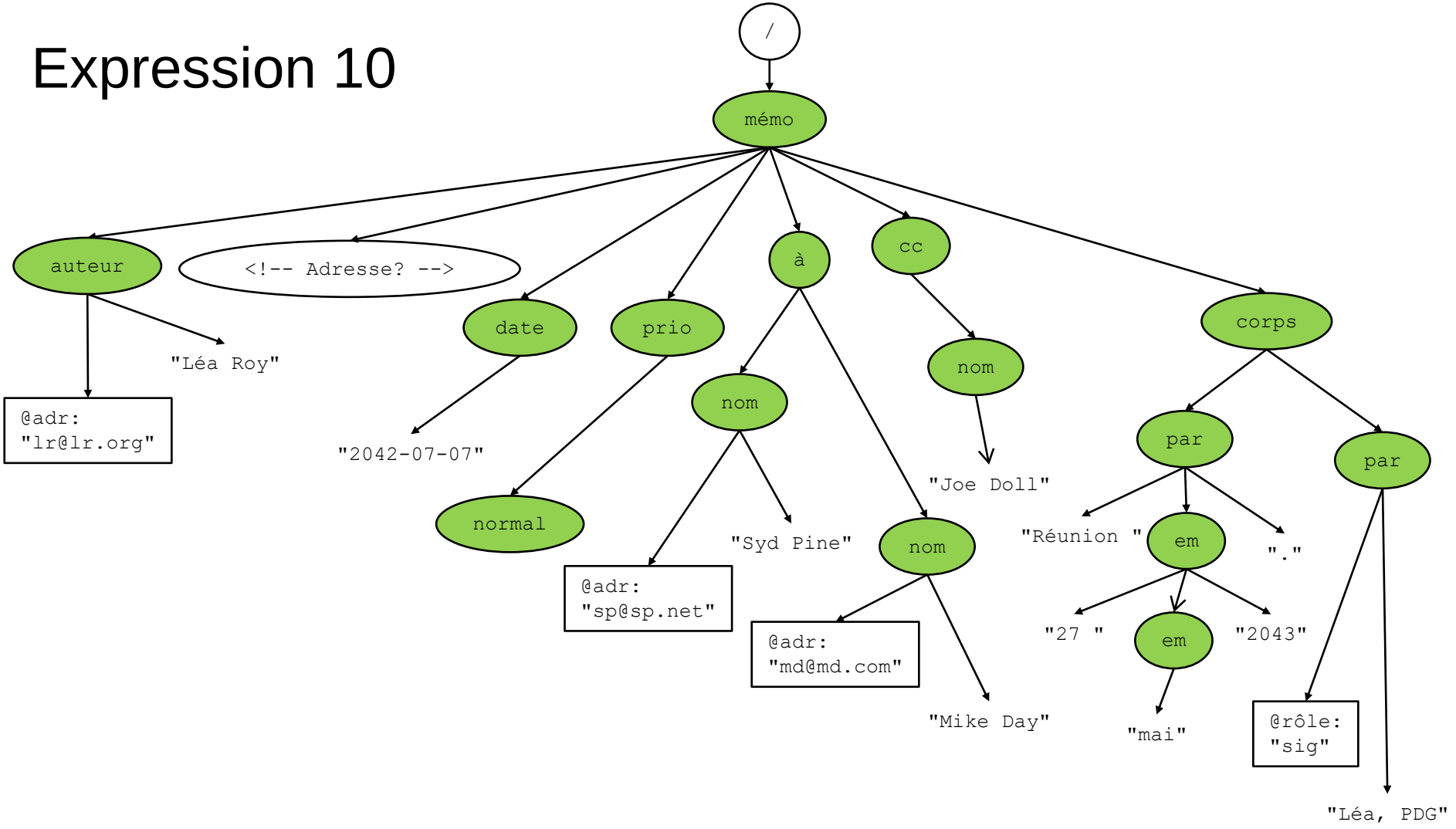
Expression: `//*[last()]`

Expression 10



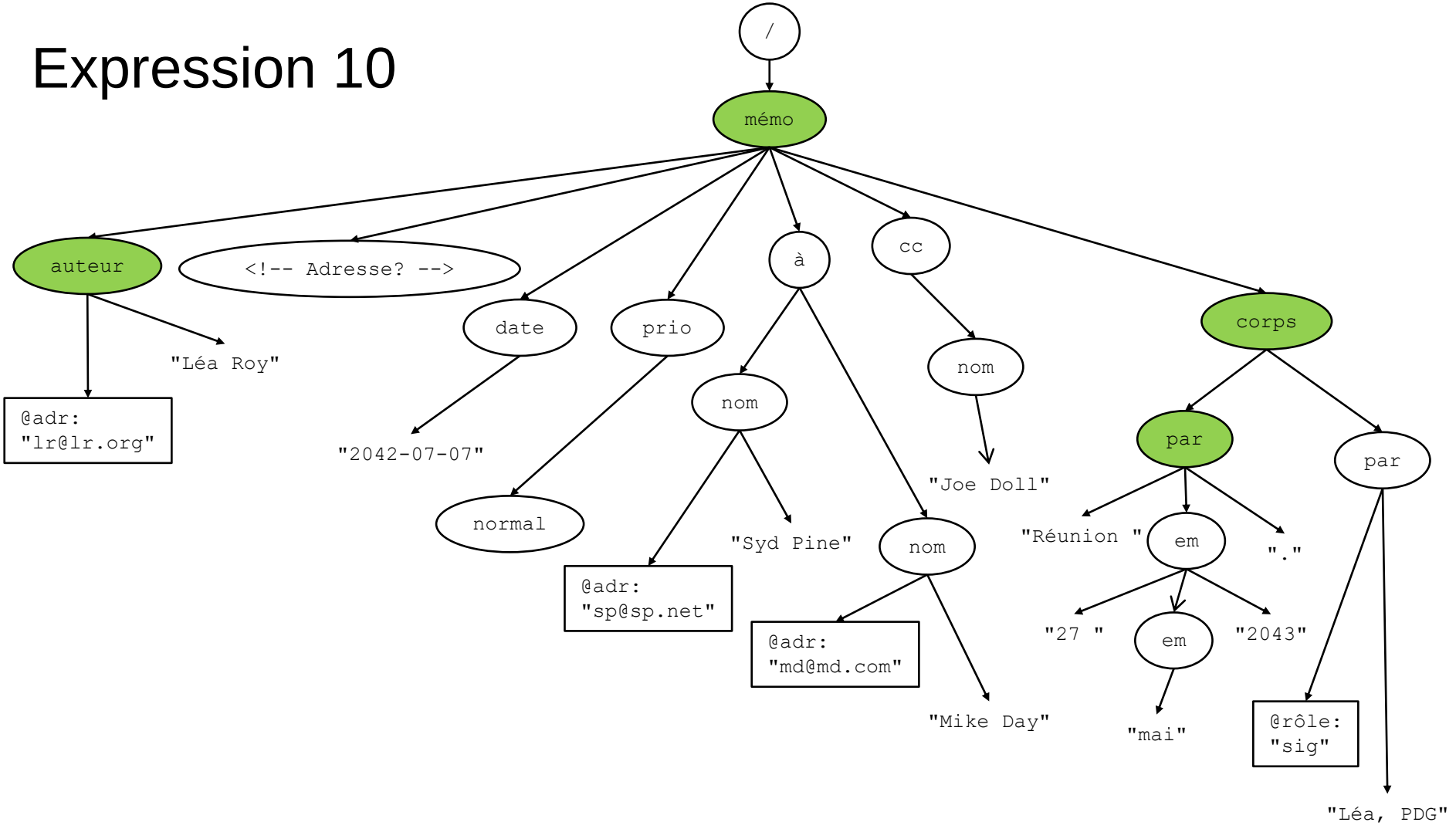
Expression: `//*[contains(.,'R')]`

Expression 10



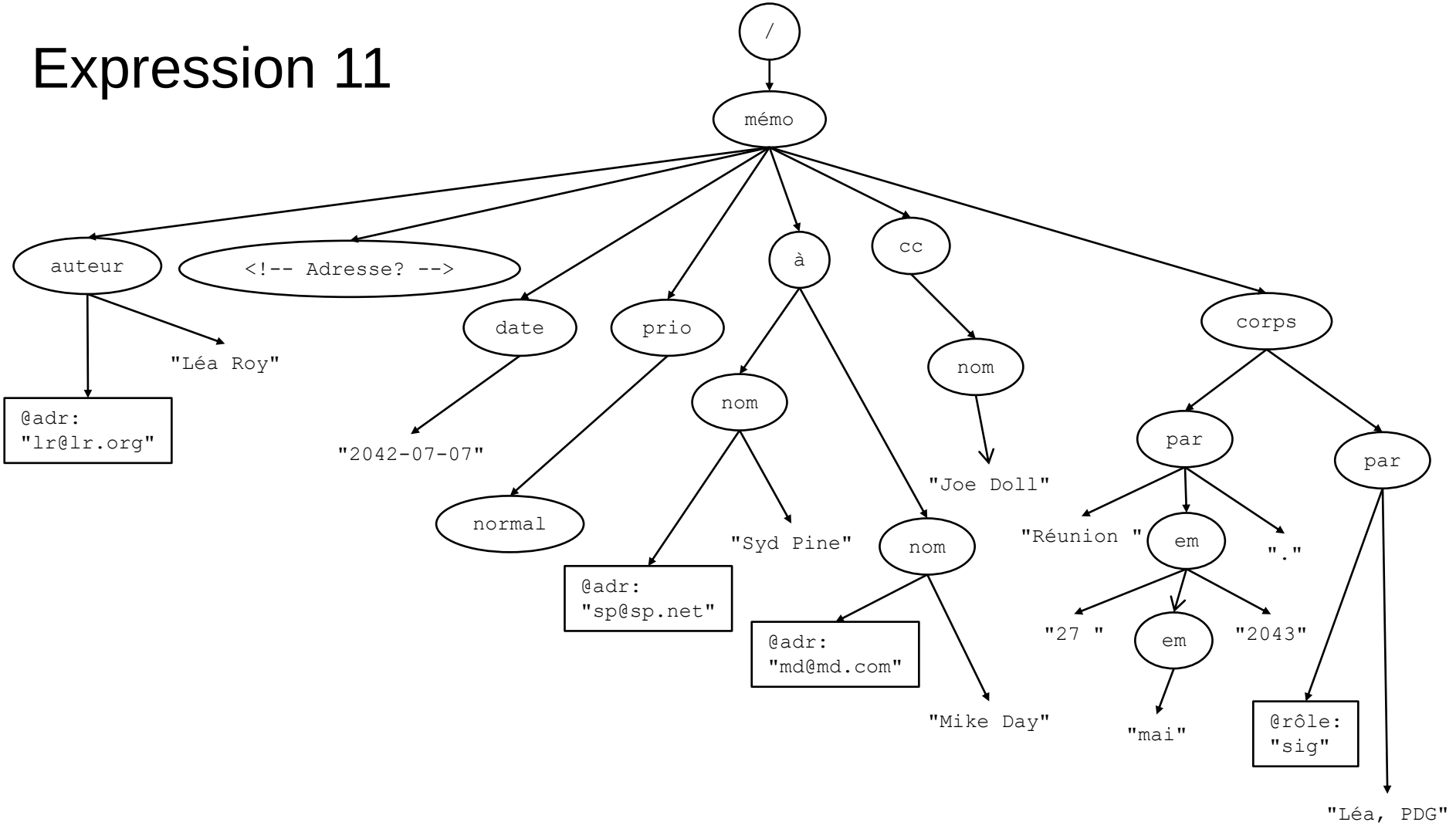
Expression: `//*[contains(.,'R')]`

Expression 10



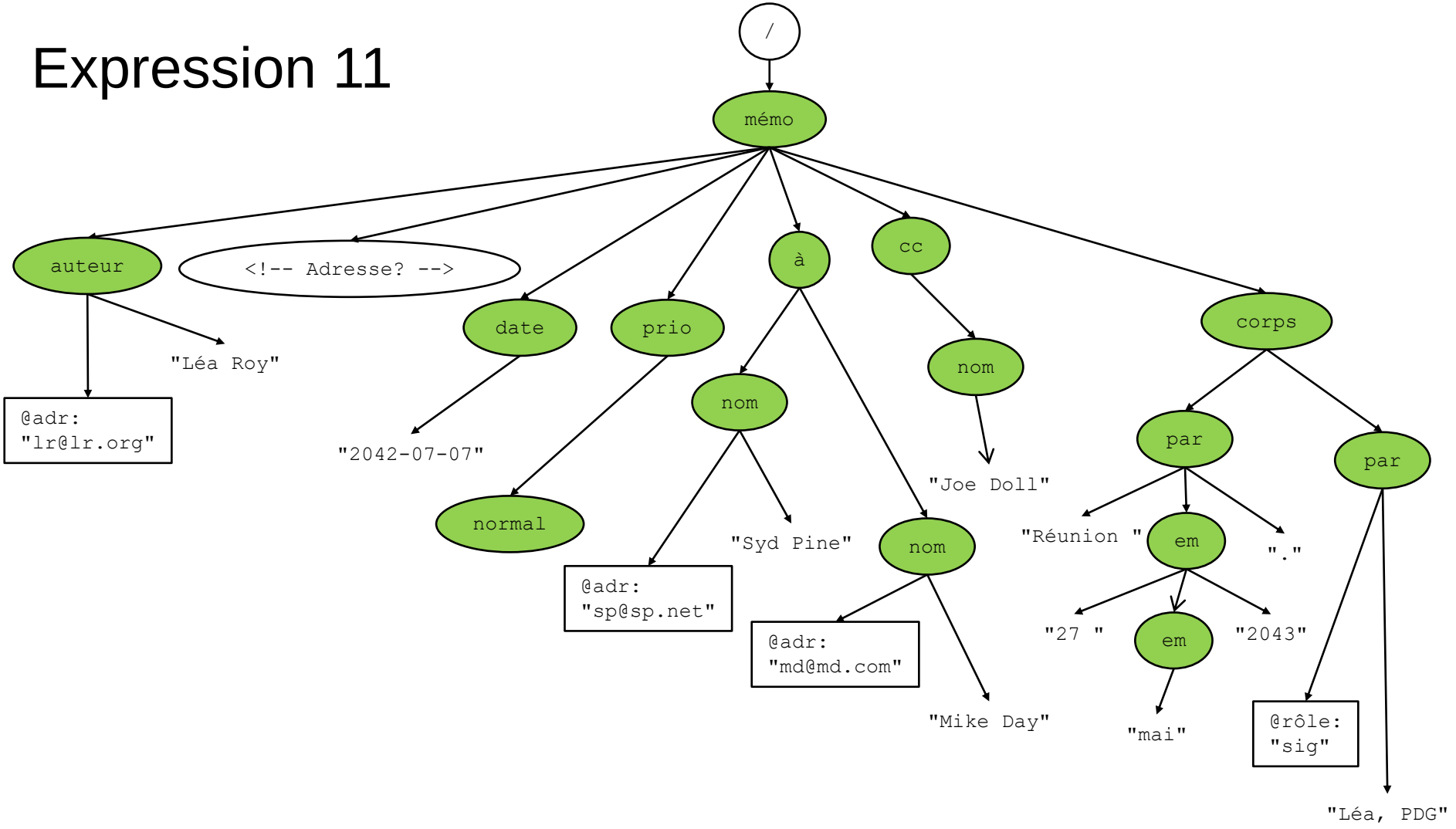
Expression: `//*[contains(.,'R')]`

Expression 11



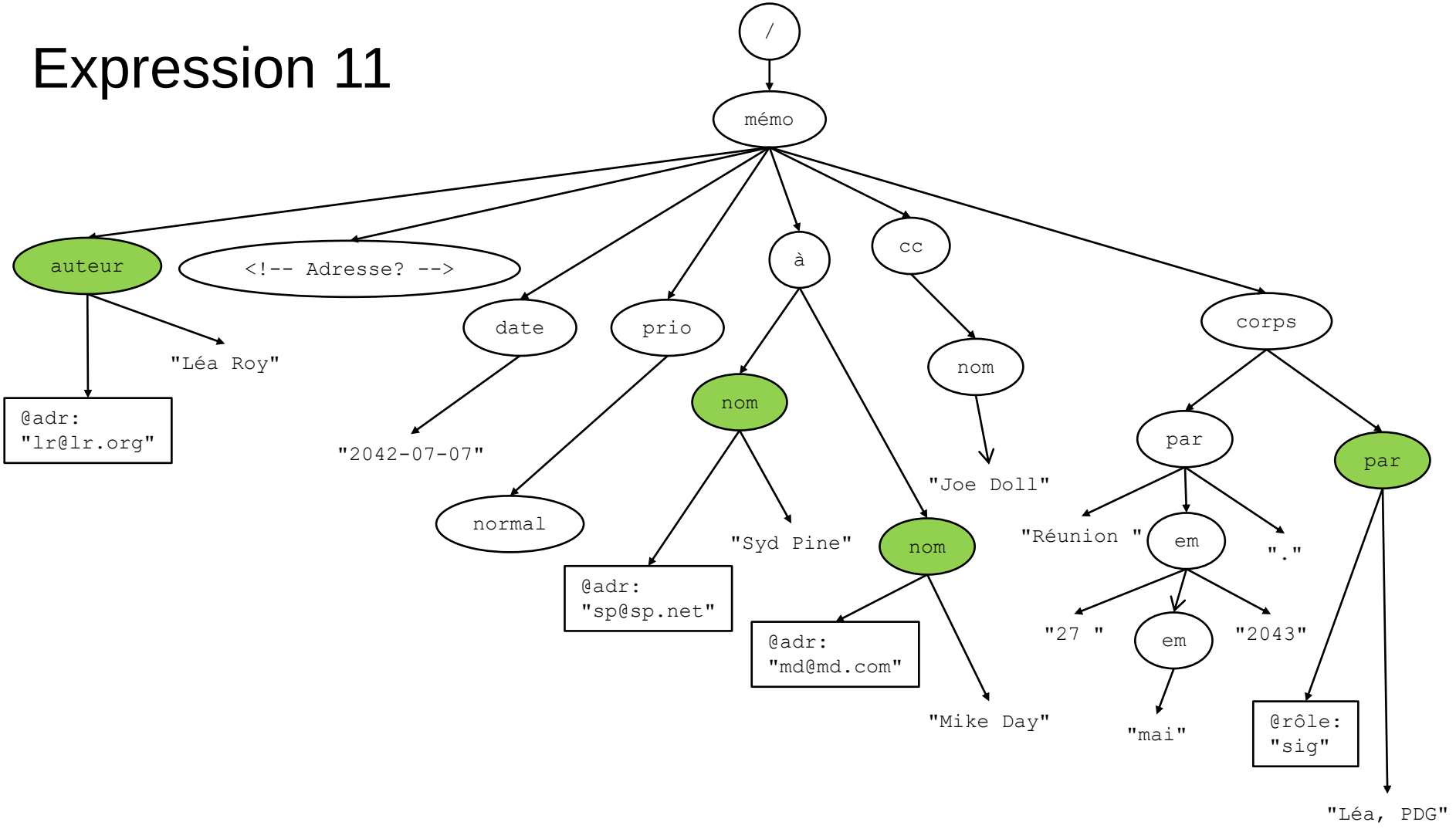
Expression: `//*[@*]`

Expression 11



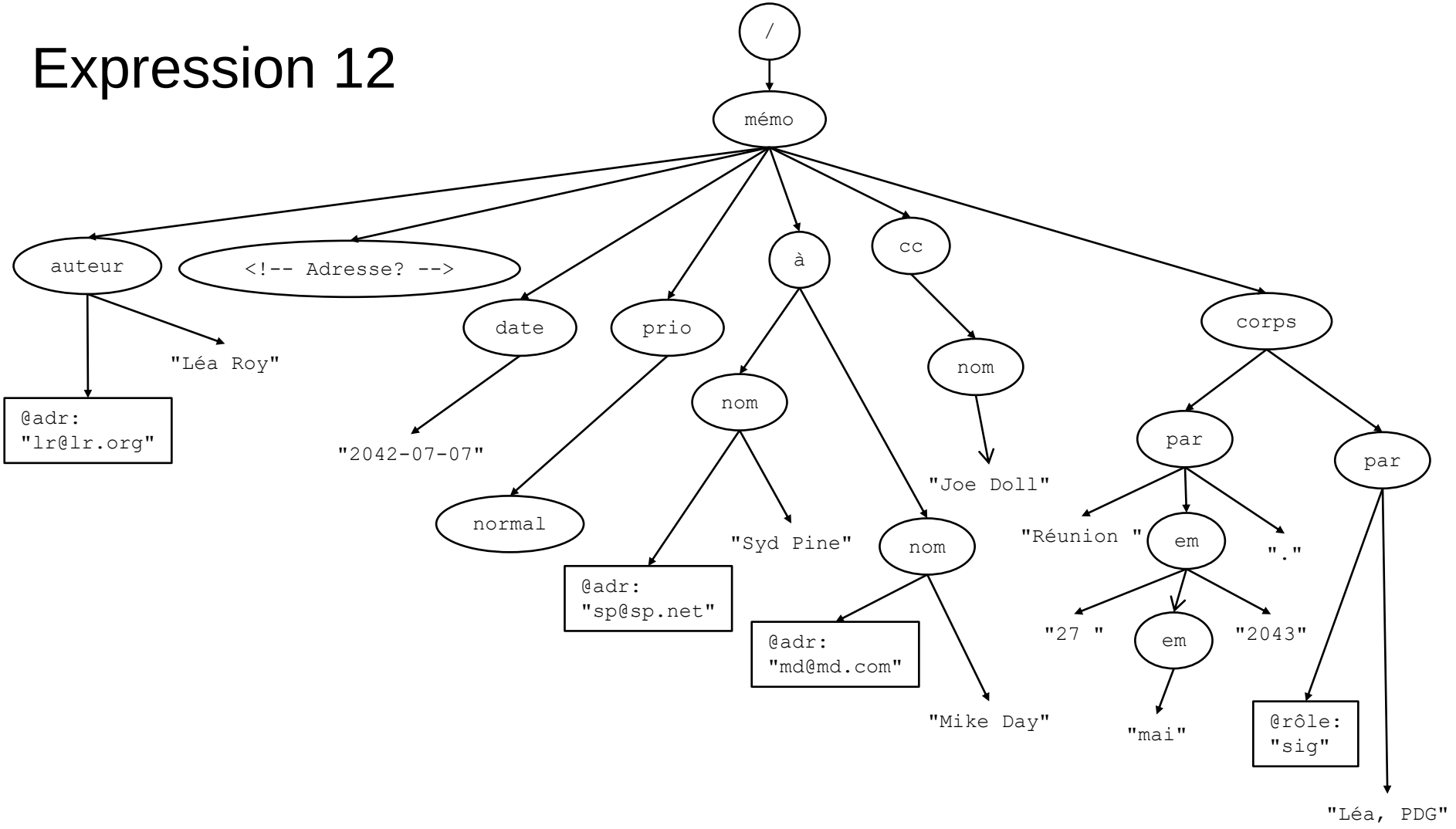
Expression: `//*[@*]`

Expression 11



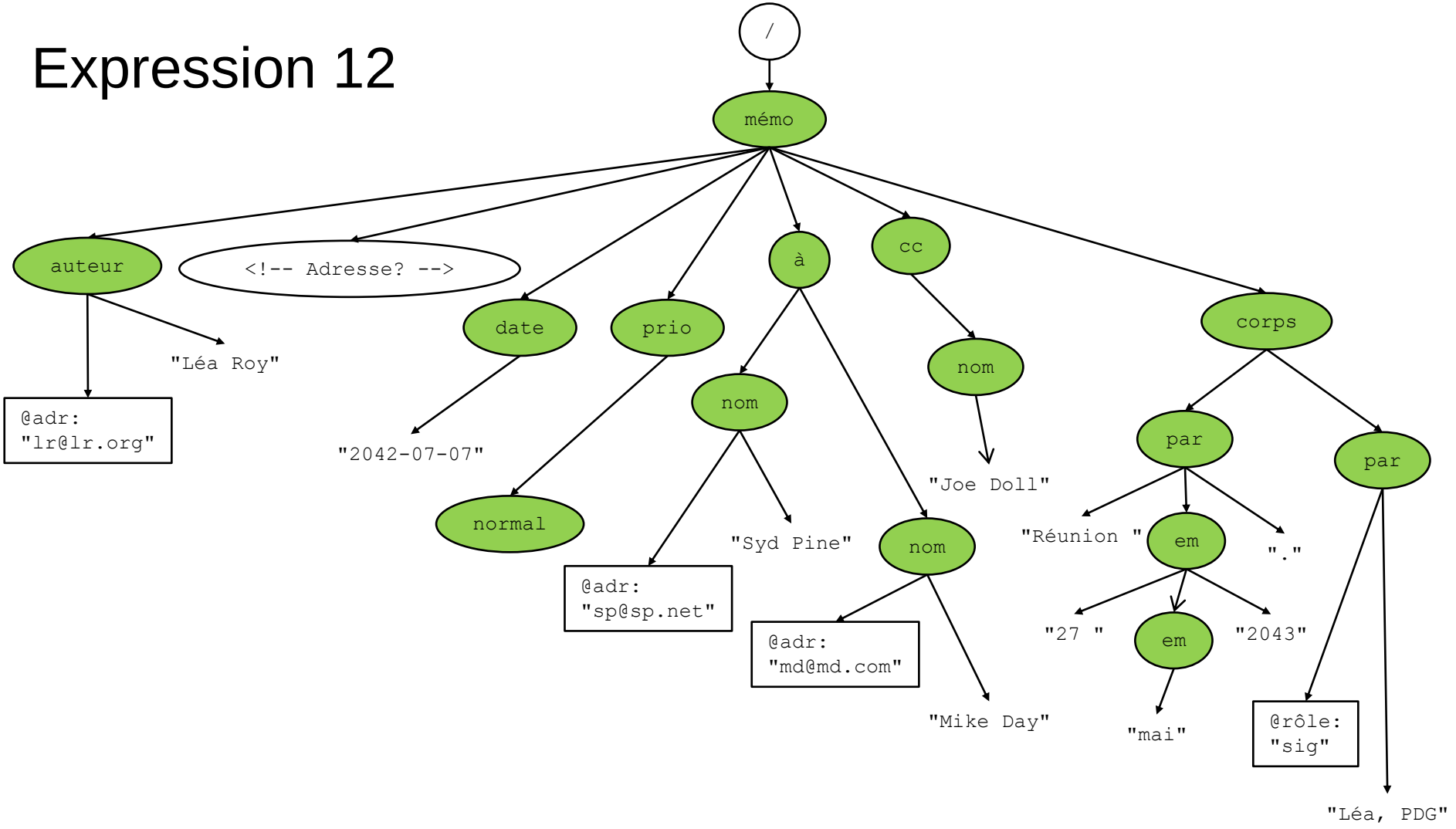
Expression: `//*[@*]`

Expression 12



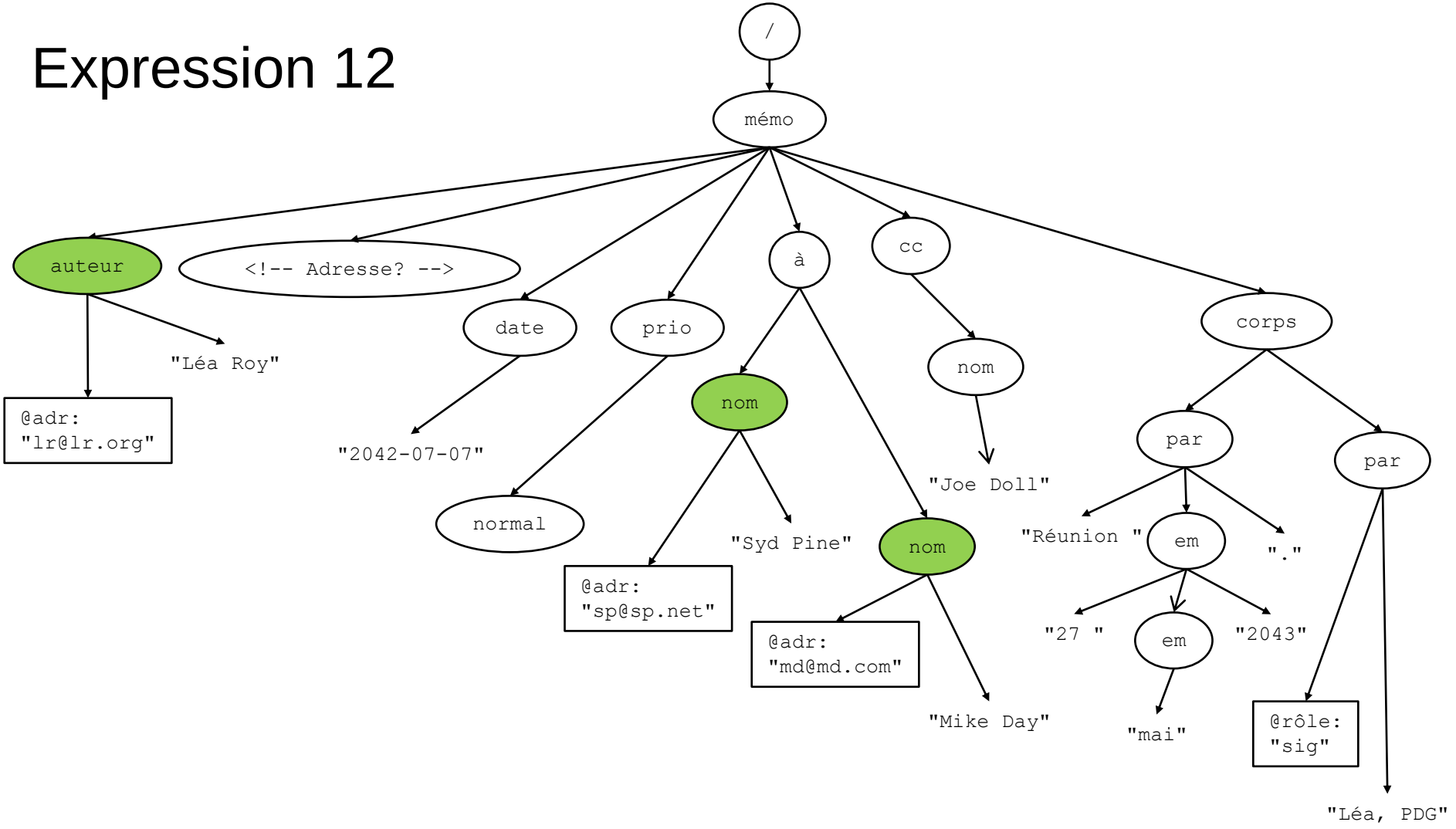
Expression: `//*[@adr]`

Expression 12



Expression: `//*[@adr]`

Expression 12



Expression: `//*[@adr]`