

condition

Soumis par Philippe YONNET

17-11-2007

Dernière mise à jour : 17-11-2007

Syntaxe

`<condition>...</condition>` (condition à predicats multiples)

syntaxe 2 :

`<condition name="X" value="Y">...</condition>` (condition à prédicat unique)

syntaxe 3 :

`<condition name="X" value="Y">...</condition>` (condition de type bloc)

AVERTISSEMENT : cette balise est à réserver aux utilisateurs avancés, tant les problèmes d'implémentations sont importants. Voir aussi les remarques en fin de description... Description

La balise condition sert à sélectionner, dans une série de réponses possibles contenues dans un template, celle qui est la plus appropriée compte tenu du contexte. Les conditions permettent d'effectuer des tests sur les valeurs actuelles d'un ou plusieurs prédicats. Il existe trois types de condition :

la condition de type bloc `<condition name="X" value="Y">` : cette syntaxe permet de tester directement si le prédicat dont le nom est X a bien pour valeur Y, si oui, le code AIML contenu entre balise fermante et balise ouvrante sera évalué.

Cette syntaxe sert à produire un branchement conditionnel unique la condition de type prédicat unique `<condition name="X">` : cette syntaxe permet de tester les valeurs du prédicat dont le nom est X. Les valeurs à tester sont indiquées par une suite de balise ``, la première balise li dont la valeur correspond est évaluée et renvoyée. La syntaxe complète de la balise condition donne donc ceci :

```
<condition name="X">
<li value="1">...[AIML]...</li>
<li value="2">...[AIML]...</li>
<li value="3">...[AIML]...</li>
...
<li value="99">...[AIML]...</li>
<li>.....</li>
</condition>
```

La dernière balise `` contient la réponse par défaut, utilisée si aucune autre valeur ne correspond. la condition de type prédicat multiple `<condition>` : cette syntaxe permet de tester la valeur de plusieurs prédicats. Les prédicats et les valeurs à tester sont indiqués par une suite de balises ``, la première balise li dont la valeur correspond est évaluée et renvoyée.

```
<condition name="X">
<li name="X" value="1">...[AIML]...</li>
<li name="Y" value="1">...[AIML]...</li>
<li name="Z" value="oui">...[AIML]...</li>
<li name="Z" value="oui">...[AIML]...</li>
<li name="Z" value="jenesaispas">...[AIML]...</li>
...
<li>.....</li>
</condition>
```

La dernière balise `` contient la réponse par défaut, utilisée si aucune autre valeur ne correspond. Attention à la syntaxe des balises `` en fonction du type de condition utilisé. On peut placer du code avant et après une balise condition, et placer plusieurs conditions dans le même modèle de réponse. Par contre, on ne peut pas imbriquer les conditions. Remarque : avec le langage AIML, il est impossible de tester si un prédicat est égal à NULL. Ce qui signifie des surprises quand on teste des variables non définies. La solution préconisée est de prévoir une valeur par défaut pour chaque prédicat utilisé, mais toutes les interpréteurs ne le permettent pas. Remarque : les pandorabots permettent des syntaxes de type `<li value="*">`. Ce qui permet de tester toute valeur non nulle d'un prédicat. Exemple d'utilisation de plusieurs conditions de type "bloc"

```
<category>
<pattern>Je suis chatain</pattern>
<template>Vous me semblez
<condition name="sexe" value="masculin"> plutôt séduisante.</condition>
<condition name="sexe" value="feminin"> plutôt bel homme.</condition>
</template>
</category>
```

Exemple d'utilisation d'une balise condition à prédicat unique

```
<category>
<pattern>Je suis chatain</pattern>
<template>Vous me semblez
<condition name="sexe">
<li value="masculin"> plutôt séduisante.</li>
<li value="feminin"> plutôt bel homme.</li>
```

```

<li> avoir toutes les qualités pour me plaire</li>
</condition>
</template>
</category>

```

Remarque : on peut donc utiliser l' ou l' autre des syntaxes en fonction des besoins. Le deuxième exemple montre que cette syntaxe permet une réponse par défaut, ce qui n'est pas le cas de la balise de type "bloc". Exemple d'utilisation d'une balise condition à prédicats multiples

```

<category>
<pattern>Je suis chatain</pattern>
<template>Vous me semblez
<condition>
<li name="categorie" value="enfant"> avoir de beaux cheveux</li>
<li name="sexe" value="masculin"> plutôt séduisante.</li>
<li name="sexe" value="feminin"> plutôt bel homme.</li>
<li> avoir toutes les qualités pour me plaire</li>
</condition>
</template>
</category>

```

Remarque : ce dernier exemple montre que cette syntaxe permet de tester plusieurs valeurs de prédicats différents pour choisir la bonne réponse. L'ordre des est important. Le premier rencontré dont la valeur correspond est évalué et retourné, et pas les suivants. Dans ce cas, le test "categorie=enfant" ne serait pas effectué systématiquement s'il n'était pas placé en première position. Pour les mêmes raisons, la balise contenant la réponse par défaut doit se situer à la fin de la liste. Compatibilité Balise utilisable dans : Modèles de réponse Conforme AIML 1.0.1 : Oui Pandorobot : Oui, avec des spécificités Référence doc : 7.3.1.