

L'attribut `namelist` indique la liste des variables à retourner au dialogue appelant, sous la forme d'objet ECMAScript. Par défaut, aucune valeur n'est retournée (contrairement à l'attribut `namelist` de la balise `<submit>` que nous verrons plus loin, et qui renvoie, par défaut, toutes les variables d'un formulaire).

On peut également utiliser l'attribut `event` afin d'envoyer un événement particulier, après être retourné au dialogue appelant.

L'ensemble de ces balises est repris dans les deux exemples ci-dessous. L'un est une illustration d'un sous-dialogue interne mis en œuvre afin de lui déléguer une tâche de recherche de parité. L'autre présente un appel de sous-dialogue externe au fichier VoiceXML principal.

Dans le programme 4.30, on déclare un sous-dialogue que l'on nomme `testeur_de_parity`. En employant le caractère `#`, on indique que la source du `<subdialog>` est interne et qu'elle se situe au niveau du formulaire nommé `calcul_de_parity`, dans le fichier courant. Dans ledit formulaire, on déclare avec la balise `<var>` l'argument `argument_un`, qui doit reprendre exactement le même nom que celui donné pour le passage de paramètre *via* la balise `<param>`. On déclare également la variable `paritee_obtenue` que l'on initialise à « pair ». Le test de parité consiste ici à évaluer le reste de la division euclidienne par 2, de l'argument passé en paramètre du sous-dialogue. Si ce reste est 1, on assigne la variable `paritee_obtenue` à « impair ». Ce résultat est renvoyé au dialogue appelant *via* la balise `<return>`.

```
<?xml version="1.0" encoding="iso-8859-1"?>
<vxml version="1.0" >
<!-- dialogue_avec_sous_dial_parity.vxml -->
<form>
```

```
  <field name="nombre_a_tester" type="number">
    <prompt>
      Choisissez un nombre ?
    </prompt>
  </field>
```

```
  <subdialog name="testeur_de_parity" src="#calcul_de_parity">
    <param name="argument_un" expr="nombre_a_tester"/>
  </subdialog>
```

```
  <block>
    <prompt>
      Le nombre <value expr="nombre_a_tester"/> est <value
      expr="testeur_de_parity.paritee_obtenue"/>.
    </prompt>
    <clear/>
  </block>
</form>
```

4.7 Sous-dialogues

```
<!-- Le sous-dialogue interne est défini ici -->
<form id="calcul_de_parity">
  <var name="argument_un"/>
  <var name="paritee_obtenue" expr="pair"/>

  <block>
    <if cond="argument_un % 2 == 1">
      <assign name="paritee_obtenue" expr="impair"/>
    </if>
    <return namelist="paritee_obtenue"/>
  </block>
</form>
</vxml>
```

Programme 4.30 Dialogue principal faisant appel à un sous-dialogue interne

Le dialogue 4.14 montre une série de tests de parité effectués grâce au programme 4.30.

```
C: Choisissez un nombre ?
H: huit cent trente-deux
C: Le nombre huit cent trente-deux est pair.
C: Choisissez un nombre ?
H: mille quatre cent vingt-sept
C: Le nombre mille quatre cent vingt-sept est impair.
C: Choisissez un nombre ?
H: vingt-trois
C: Le nombre vingt-trois est impair.
C: Choisissez un nombre ?
H: huit cent quatre-vingt-huit
C: Le nombre huit cent quatre-vingt-huit est pair.
C: Choisissez un nombre ?
H: zéro
C: Le nombre zéro est pair.
C: Choisissez un nombre ?
H: mille trente-deux
C: Le nombre mille trente-deux est pair.
```

Dialogue 4.14 Calcul de parité grâce à un sous-dialogue interne

Le programme 4.31 illustre un mécanisme de sous-dialogue externe. D'abord, le système demande à l'utilisateur d'entrer deux nombres, afin de passer à leur addition. Le calcul ne se fait pas au sein de ce fichier, mais par l'intermédiaire du programme 4.32 qui fait office de sous-dialogue pour notre dialogue principal, et dont la tâche consiste à additionner les deux arguments passés en paramètre, puis à renvoyer à l'appelant la somme calculée.

```
<?xml version="1.0" encoding="iso-8859-1"?>
```