

REALISATION DU NOYAU COMPREHENSION-GESTION DE DIALOGUE

Développement et Test d'une application VoiceXML

1- Cadre de l'application

On envisage de développer un système de dialogue oral permettant aux personnes abonnées d'accéder à différents services comme la gestion d'un agenda, la gestion d'un répertoire, ... Ainsi, un abonné pourra contacter le serveur par téléphone et dialoguer avec le système pour que la tâche envisagée soit effectuée.

Vous serez chargé de développer la version V1 du système de dialogue. Cette version ne concerne que l'écriture et le test du noyau chargé de la **compréhension et de la gestion du dialogue** compte tenu de la tâche à réaliser.

Dans un premier temps, les entrées sorties vocales seront simulées et seul le texte écrit sera traité. Le noyau est constitué d'un ou plusieurs documents VoiceXML (fichiers .vxml) et de grammaires appropriées (fichiers .grxml).

Tâche à réaliser : identifier la requête à exécuter (= transaction à effectuer vers une Base de données). En effet l'agenda est supposé être mémorisé sous forme d'une base de données qui peut être consultée ou modifiée par le biais de requêtes SQL (**insert**, **update**, **select**, **delete**). La base de données sera simulée, seuls les éléments nécessaires pour effectuer la requête seront extraits des différentes phases du dialogue avec l'utilisateur.

Un noyau très primitif (V0) vous sera fourni. Vous le ferez évoluer pour que les objectifs énoncés ci-dessous soient atteints. Des exemples de dialogue sont **donnés en annexe** pour vous donner une idée de scénarios possibles.

2- Composants à utiliser :

Utiliser le logiciel **Optimtalk** (sous Windows), interpréteur VXML (VoiceXML).

1) Vous utiliserez principalement 2 outils :

- **ot_grammar_tester.exe** : cet outil permet de tester une grammaire écrite en langage SRGS (syntaxe Speech Recognition Grammar Specifications) fichier avec extension .grxml.

```
ot_grammar_tester.exe ma_grammaire.grxml
```

- **ot_vxml_interpreter.exe** : cet outil permet de lancer une application VoiceXml (fichier extension .vxml) pour la tester « offline » c'est-à-dire sans la reconnaissance de la parole et sans la synthèse (interface texte).

```
ot_vxml_interpreter.exe mon_appli.vxml
```

- 2) Ces outils se trouvent dans le répertoire indiqué en début de TP. Pour les exécuter (version basique) il faut ouvrir une fenêtre de commande DOS (cmd) se placer dans le répertoire mentionné et indiquer le chemin menant à la grammaire ou l'application à tester.
- 3) Les messages d'erreurs n'étant pas très explicites dans la version basique, vous pouvez consulter les fichiers de log qui sont mis à jour à chaque exécution de l'interpréteur. Ces fichiers se trouvent dans le répertoire de l'application.
- 4) Un document pdf annexe concernant la syntaxe des fichiers .grxml et .vxml est disponible sous moodle. Vous vous appuyerez également sur le jeu d'exemples contenus dans le dossier « Exemples ».

3- TP et BE : Cahier des charges

I- TP n°1 :

Objectif n°1 = Récupérer la version V0 fournie et tester les grammaires.

- a) Vous utiliserez d'abord l'outil **ot_grammar_tester.exe** pour tester chacune des grammaires fournies, comprendre ce qu'elles font et comment cela est implémenté.
 - grammaire_nombre_v3.grxml
 - grammaire_num_ab.grxml
 - grammaire_act_lang_confirmation.grxml
 - grammaire_aide.grxml
 - grammaire_dates_v3.grxml
- b) Vous testerez votre grammaire (devoir n°1) et notamment la possibilité de reconnaître un **horaire** quelconque énoncé de différentes façons : *midi moins le quart, onze heures quarante-cinq...*

- c) Vous modifierez la grammaire pour renvoyer le texte reconnu et extraire le sens associé à l'horaire reconnu (valeurs numériques correspondantes) constitué de trois champs :
H : 11 MN : 45 NB_MN : 111
- d) Vous ferez les modifications nécessaires au bon fonctionnement de celle-ci et ferez valider la grammaire par l'enseignant. La version **finalisée et validée** de votre grammaire devra être déposée sur moodle (ex : dupond_horaire_valid.grxml) ainsi que des exemples d'exécution(ex : dupond_horaire_test.txt).
- e) Utilisez ensuite l'outil `ot_vxml_interpreter.exe` pour exécuter la version V0 de l'application qui vous est fournie et analyser le contenu du fichier vxml.
- f) Ajout d'une contrainte d'application : pour simuler le comportement du système en situation de code erroné, vous modifierez la gestion de dialogue pour qu'un contrôle de validité du numéro d'abonné soit réalisé (numéro compris entre 1 et 2500 per exemple).

II- TP n°2 :

Objectif n°2 = Ecrire une nouvelle version V1 à partir de la V0 fournie

Cette version permettra à un abonné d'enregistrer un nouveau rendez-vous. On considérera qu'un rendez-vous est caractérisé par une **date**, un **horaire**, un **lieu** et une **personne** éventuellement un **objet**.

- g) Modifiez la gestion de dialogue pour prendre en compte la **date** de rendez-vous. Utilisez pour cela la grammaire externe fournie et demandez une **confirmation explicite** de l'information.
- h) Modifiez la gestion de dialogue pour intégrer l'information sur l'**horaire** et demander une **confirmation explicite** de l'horaire.

ATTENTION : les messages du système étant destinés à être envoyés à un module de synthèse vocale, **le message généré doit être compréhensible par l'utilisateur**. Différencier les messages de TRACE, des messages correspondant à la réponse du système en situation d'interaction. .

- i) Vous veillerez ensuite à vérifier la **validité de l'horaire** extrait en prenant en compte les contraintes d'application suivantes :
 - un horaire valide est compris entre 00h et 23h59 (75 heures n'est pas un horaire valide)
 - dans cette application, les rendez-vous ne peuvent être pris qu'entre 7h30 et 21h30

- j) Modifiez la gestion de dialogue pour pouvoir **sortir de l'application à tout moment**.

III- TP n°3 :

Objectif : continuer la version V1

- a) Définir les grammaires nécessaires pour identifier le **lieu** de rendez-vous, la **personne** et l'**objet** de celui-ci et extraire l'information nécessaire à l'application. Testez ces grammaires.
- b) Intégrer les demandes d'informations correspondantes dans la partie gestion de dialogue. Les demandes se feront suivant une stratégie d'**interaction directive et une stratégie de confirmation explicite**. Dans chaque cas les messages d'erreur ou d'aide seront adaptés au contexte.
- c) Transformer la séquence de confirmation explicite en sous-dialogue. (voir poly VXML pour plus de détails rubrique **<subdialog>**).

IV- TP n°4, 5 et suivante (TP/BE)

Objectif : finaliser la version 1 et développer la version 2

- a) Transformer la séquence de recueil des informations caractérisant un rendez-vous en un sous-dialogue (voir poly VXML pour plus de détails rubrique **<subdialog>**).

A partir de cette version V1, écrire une **seconde version V2** dans laquelle l'utilisateur aura le choix de procéder à 5 actions différentes :

- (1) **ajouter** un nouveau rendez-vous (déjà fait en fait puisqu'il faut recueillir toutes les informations caractérisant un rendez-vous) ;
- (2) **consulter** les rendez-vous suivant un ou plusieurs critères : *consultation suivant le jour, le lieu ou la personne par exemple*
- (3) **modifier** un rendez-vous (supposé existant) suivant un ou plusieurs critères : *lieu, objet, personne*, ou le **déplacer** en changeant le jour et/ou l'heure.
- (4) **supprimer** un rendez-vous suivant un ou plusieurs critères ;

L'interprétation à associer à chacune des actions sera liée au type de requête SQL à exécuter pour réaliser la transaction demandée :

Ajouter =>	INSERT	Consulter =>	SELECT
Modifier =>	UPDATE	Supprimer =>	DELETE

Les critères correspondent à l'information connue sur la base de laquelle la transaction va s'effectuer (clause **WHERE** ...)

Dans cette version vous privilégieriez une stratégie d'**interaction mixte** :

- les demandes d'information associées à chaque action possible se feront par le biais de **sous-dialogues**
- la **stratégie de confirmation** sera **implicite** par défaut et pourra être amenée à changer suivant la répétition des cas d'erreurs
- le sous-dialogue associé à l'action **ajouter** devra permettre la **prise en compte d'énoncés contenant plusieurs informations**.

« Je veux ajouter un rendez-vous mardi neuf novembre à dix heures trente à l'AIP »

- à tout moment l'utilisateur pourra **demandeur de l'aide** au système (« Qu'est-ce que je peux faire ? ... »)

Attention : Il n'y a pas de bases de données à gérer. L'objectif est de permettre au système de recueillir toutes les informations nécessaires auprès de l'utilisateur de façon à construire la requête SQL. Il n'y a pas d'interaction avec une vraie BD.

V- Evaluation :

Vous déposerez sous moodle **AVANT le 20/01/2016** :

- une **archive** (à votre nom de binôme) contenant l'intégralité du code commenté de la version 2 en vue d'une validation sur machine ;
- ainsi qu'un **rapport** décrivant l'organisation de votre application en terme de **concepts**, de **stratégies** (stratégie d'interaction et de confirmation) et **modèles de dialogue** associés à chaque sous-tâche implémentée. Vous donnerez également quelques exemples d'exécution et de tests validant les aspects du travail réalisé (flexibilité, gestion des erreurs, gestion de l'aide, sortie à tout moment, confirmation, infirmation, stratégies de gestion de dialogue, ...).

Exemples de dialogue (indicatif)

DIALOGUE 1 :

S : Bonjour et Bienvenue sur le service de gestion des agendas. Quel est votre numéro d'abonné ?

U : c'est le cent cinquante trois ?

S : Voulez-vous accéder à l'agenda de l'abonné numéro cent cinquante trois ?

U : Oui c'est cela.

S : Quel jour souhaitez-vous consulter ?

U : le mercredi vingt huit février

S : Souhaitez-vous consulter votre agenda pour le mercredi vingt huit février deux mille onze ?

U : eh bien oui

S : Quelle période vous intéresse ?

U : le matin

S : Vous voulez des informations concernant vos rendez-vous prévus entre sept heures trente et douze heures trente ?

U : oui oui.

S : Ne quittez pas je recherche votre information ...

DIALOGUE 2 :

S : Bonjour et Bienvenue sur le service de gestion des agendas. Quel est votre numéro d'abonné ?

U : c'est le ... je cherche ah oui le cent cinquante trois ?

S : Excusez-moi je n'ai pas compris. Pouvez-vous me redonner votre numéro d'abonné ?

U : un cinq trois

S : Voulez-vous accéder à l'agenda de l'abonné numéro cent cinquante trois ?

U : Oui c'est cela.

S : Quel jour souhaitez-vous consulter ?

U : jeudi trente février

S : Cette date n'est pas valide. Pouvez-vous me préciser la date qui vous intéresse ?

U : demain.

S : Souhaitez-vous consulter votre agenda pour le mercredi dix novembre deux mille dix ?

U : c'est ça

S : Quel période vous intéresse ?

U : après trois heures

S : Vous voulez des informations concernant vos rendez-vous prévus à partir de quinze heures ?

U : oui c'est ça.

S : Ne quittez pas je recherche votre information ...

DIALOGUE 3 :

S : Bonjour et Bienvenue sur le service de gestion des agendas. Quel est votre numéro d'abonné ?

U : trente six ?

S : Quelle opération souhaitez-vous effectuer sur l'agenda de l'abonné numéro trente six ?

U : je voudrais déplacer un rendez-vous.

S : Quel jour est prévu le rendez-vous à déplacer ?

U : le dix sept novembre à dix heures.

S : Quand souhaitez-vous reporter le rendez-vous prévu le mercredi dix sept novembre deux mille dix ?

U : je veux le déplacer à deux heures

S : le rendez-vous prévu à neuf heures doit-il être déplacé le même jour à quatorze heures ?

U : oui

S : Ne quittez pas je procède à la modification ...

DIALOGUE 4 :

S : Bonjour et Bienvenue sur le service de gestion des agendas. Quel est votre numéro d'abonné ?

U : dix mille trente six ?

S : le numéro dix mille trente six n'est pas valide. Veuillez redonner votre numéro ?

U : sept mille neuf cents

S : le numéro sept mille neuf cents n'est toujours pas valide. Veuillez procéder à un troisième et dernier essai ?

U : vingt mille deux

S : Nous n'avons pas pu reconnaître votre numéro d'adhérent. Vous ne pouvez donc utiliser ce service. Au revoir.

DIALOGUE 4 :

S : Bonjour et Bienvenue sur le service de gestion des agendas. Quel est votre numéro d'abonné ?

U : douze ?

S : Vous êtes l'abonné numéro douze. Quelle opération souhaitez vous faire ?

U : je ne sais pas ce que je peux faire

S : Vous pouvez ajouter, modifier, supprimer ou consulter vos rendez-vous ou bien quitter ce service. Que souhaitez-vous faire ?

U : je veux partir

S : Vous quittez le service de gestion des agendas. Au revoir.