La standardisation des applications vocales

David Janiszek
LIPADE/DIADEX
Université Paris Descartes

UNIVERSITÉ PARIS DESCARTES

Plan

- Contexte
  - Problématique
  - Historique
- Architecture
  - Serveur Vocal Interactif
  - L'approche service WEB
  - L'architecture W3C
- Normes et standards
  - RFC sous-jacentes
  - VoiceXML
  - ECMAScript
  - Speech Recognition Grammar Specification – SRGS

Plan (suite)

- Stochastic Language Models (N-Gram) Specification
- Speech Synthesis Markup Language - SSML
- Call Control XML - CCXML (suite)
- State Chart XML (SCXML) : State Machine Notation for Control Abstraction

- Compléments
  - Conception d'une application vocale
  - Les industriels du secteur
  - Les outils open-source
- Perspectives

Contexte

Problématique
Historique

Problématique

- Fonctionnelle
  - Améliorer l'accessibilité
    - « Ouvrir le web à toute personne pouvant écouter et parler »
      - Handicap
    - Téléphone vs ordinateur
  - Convergence
    - Approche plurimédia

Problématique (suite)

- Systémique
  - Le système de dialogue est conçu pour une seule application
    - Architecture
      - Logiciel spécialisé
      - Matériel propriétaire (téléphonie)
    - Modèles
  - Absence de standard
    - Intégration difficile
    - Coût de maintenance
    - Mise à jour difficile
    - Réutilisabilité limitée

Historique

Évènement	Acteur(s)	Date
Phone Markup Language (PML)	AT&T, Lucent	1995
Conférence sur les Interactive Voice Response Markup Languages	W3C	1998
Fondation du VoiceXML Forum	W3C	1998
VoxML	Motorola	1998/1999
SpeechML	IBM	1999

Historique (suite)

Évènement	Acteur(s)	Date
VoiceXML 1.0	W3C	2000
Speech Application Language Tags (SALT)	Microsoft, Cisco Systems, Comverse, Philips Consumer Electronics, Scansoft	2001-2007
SALT 1.0	id.	2002
VoiceXML 2.0	W3C	2004
VoiceXML 2.1	W3C	2007

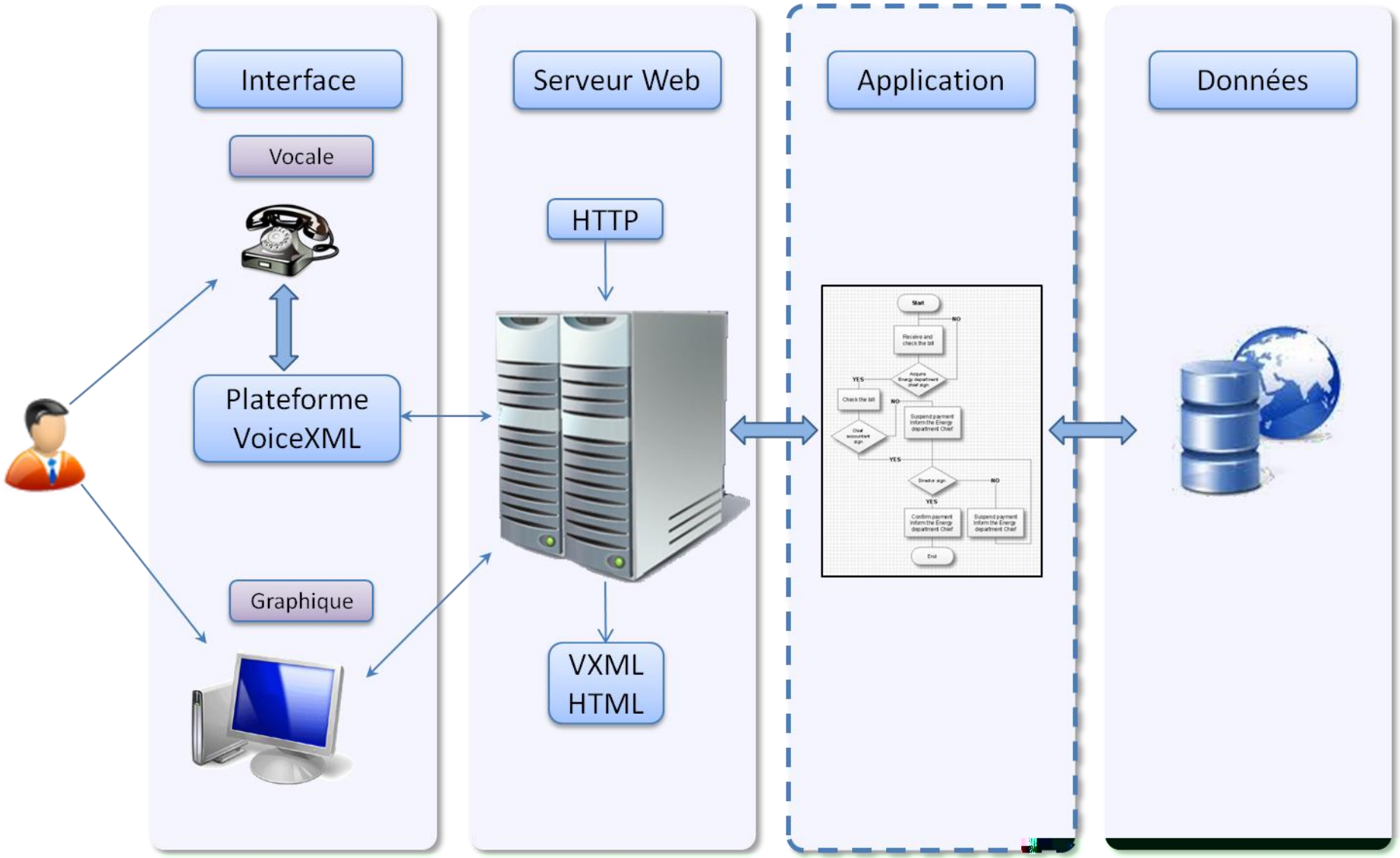
Architecture

Serveur Vocal Interactif (SVI)
L'approche service WEB
L'architecture W3C

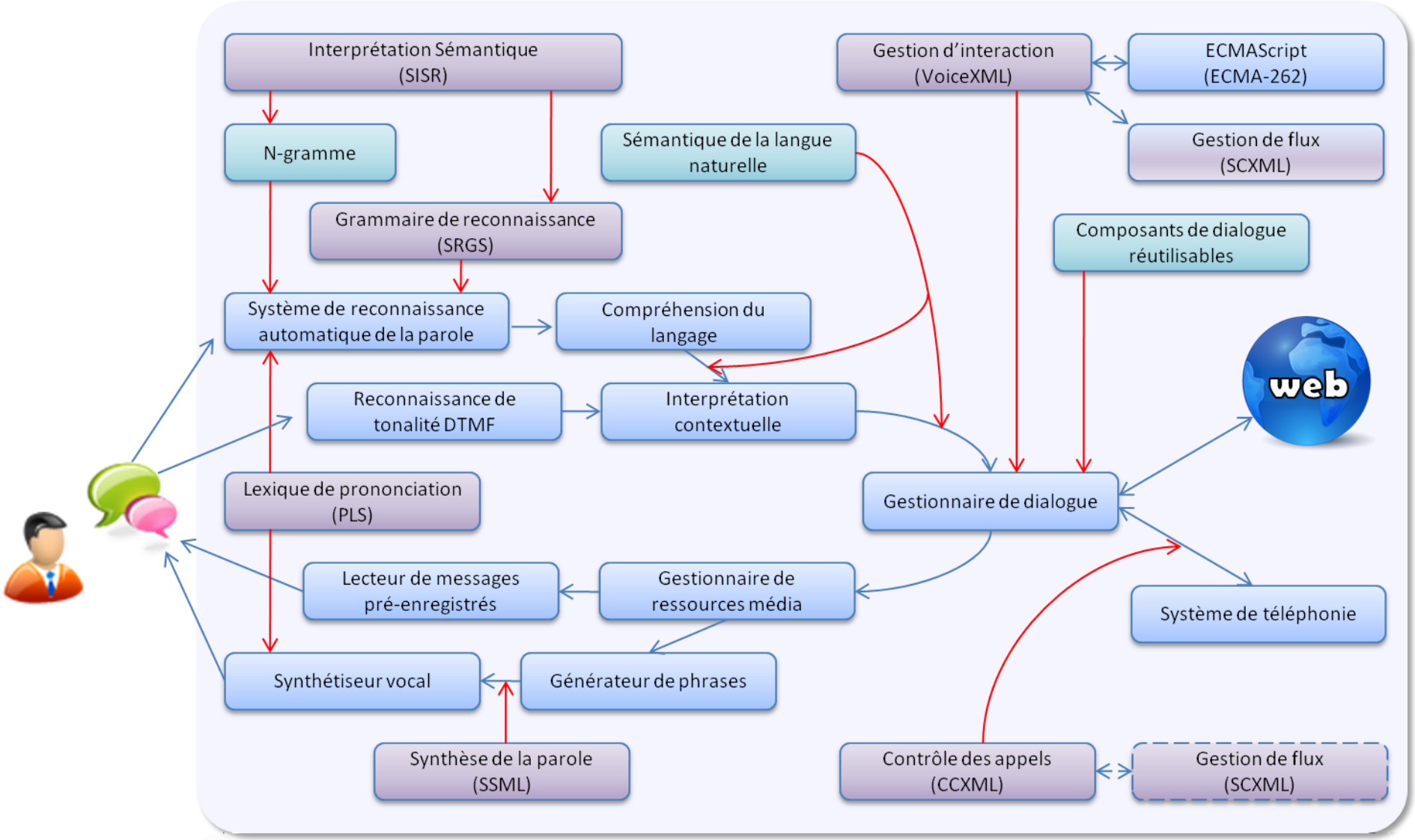
Serveur vocal interactif (SVI)

- Automatiser le traitement des appels téléphoniques
  - Interactive Voice Response (IVR)
  - Messages pré-enregistrés/DTMF → Interaction vocale
    - L'interaction est contrainte
  - Applications:
    - Communication
      - Email par téléphone, répertoire, rappel de RdV, …
    - Contenu
      - Actualités, météo, bourse, horaires d'avion, …
    - Productivité
      - Force de vente, centre d'appels, gestion clientèle, …
    - Commerce
      - Transaction banquière, réservation d'hôtel, …

L’approche service WEB



L'architecture W3C



Normes et standards

RFC sous-jacentes
VoiceXML
ECMAScript
Speech Recognition Grammar Specification – SRGS
Stochastic Language Models (N-Gram) Specification
Speech Synthesis Markup Language - SSML
Call Control XML - CCXML (suite)
State Chart XML (SCXML) : State Machine Notation for Control Abstraction

RFC sous-jacentes

- RFC 4267 : The W3C Speech Interface Framework Media Types … (2005)
  - application/voicexml+xml
  - application/ssml+xml
  - application/srgs, application/srgs+xml
  - application/ccxml+xml
  - application/pls+xml
- RFC 3023 : XML Media Types (2001)
  - charset (UTF, ISO, …)
  - DTD (Document Type Definition)
  - mathml
  - xslt (eXtensible Stylesheet Transformation)
  - rdf (Resource Description Framework)
  - svg (Scalable Vector Graphics)

VoiceXML

- Standard développé par le Voice Browsing Working Group (VBWG)
  - Le groupe de travail le plus important du W3C
- Langage de script (simple)
  - Application permettant une interaction vocale (ou DTMF)
- Séparation entre l'interface et l'application
  - Capable d'exploiter l'existant (web)
- Indépendant de la plateforme
  - SRAP
  - Téléphonique
- Pilote le moteur de reconnaissance

Exemple

Fichier VoiceXML

```
<?xml version="1.0"?>
 <vxml version="2.0" xml:lang="fr-FR">
  <property name="inputmodes" value="dtmf voice"/>
   <form id="distributeur">
    <block>
     <prompt>
       Souhaitez-vous un café, un thé, du lait ou rien ?
     </prompt>
    </block>

     <grammar src="boisson.grxml" type="application/srgs+xml"/>

    <block>
     <prompt>
       <audio src="http://www.exemple.com/audio/aurevoir.wav">
        Merci et au revoir
       </audio>
     </prompt>
    </block>
   </form>
 </vxml>
```

Interactions

S : Souhaitez-vous un café, un thé, du lait ou rien ?

U : Un jus d'orange

S : Je n'ai pas compris

S : Souhaitez-vous un café, un thé, du lait ou rien ?

U : Thé

→ joue le fichier aurevoir.wav
ou synthétise "Merci et au revoir"

Exemple (suite)

Fichier VoiceXML

```
<form id="distributeur">
 <field name="choix">
  <block>
   <prompt>
    Souhaitez-vous un café, un thé, du lait ou rien ?
   </prompt>
  </block>
  <grammar root="boisson">
   <rule id="boisson">
    <one-of>
     <item><tag>$='coffee'</tag>    café</item>
     <item><tag>$='tea'</tag>       thé</item>
     <item><tag>$='milk'</tag>      lait</item>
    </one-of>
   </rule>
  </grammar>
 </field>
</form>
```

Comportement

Variable à valuer : choix

Synthèse du message : "Souhaitez-vous un café, un thé, du lait ou rien ?"

Définition d'une grammaire locale
Définition d'une règle "boisson"

$boisson = café | thé | lait

Si on reconnait café
alors choix="coffee"
…

Exemple (suite)

Fichier VoiceXML

```
<form id="distributeur">
 <var name="temperature"/>
 <field name="choix">

 ...

  <filled>
   <if cond="choix=='milk'">
    <assign name="temperature" expr="'froid'"/>
   <else/>
    <assign name="temperature" expr="'chaud'"/>
   </if>
   <prompt>
    Vous avez demandé un
      <value expr="choix"/>
      <value expr="temperature" />
   </prompt>
  </filled>

 </field>
</form>
```

Comportement

Variable à valuer : temperature

Si l'utilisateur a demandé du lait
alors on synthétise
"Vous avez demandé un lait froid"
sinon on synthétise
"Vous avez demandé un XXX chaud"

où XXX=café| thé

ECMAScript

- European association for standardizing information and communication systems (ECMA)
- Langage de progammation standardisé (script)
- Standard ECMA-262 (1999)
- Exemples:
  - Javascript (ECMA-262/version 3)
  - Microsoft Jscript (ECMA-262/version 3)
  - Adobe ActionScript (ECMA-262/version 4)

Speech Recognition Grammar Specification - SRGS

- La grammaire est transmise au moteur de reconnaissance

- Définit:
  - Grammaire de reconnaissance hors-contexte
    - ABNF
    - XML
  - Application vocale/DTMF

- Permet:
  - Pondération des règles
  - Probabilité de répétition d'un mot

Exemple

Fichier SRGS

```
<?xml version="1.0"?>
 <grammar version="1.0“  xml:lang=“fr-FR“
   tag-format="semantics/1.0" root=“reponse">
  <rule id=“reponse" scope="public">
   <one-of>
    <item><ruleref uri="#oui"/></item>
    <item><ruleref uri="#non"/></item>
   </one-of>
  </rule>
  <rule id=“oui">
   <one-of>
    <item>oui</item>
    <item>ouais<tag>out=“oui";</tag></item>
     <item><token>bien sûr</token>
                                  <tag>out=“oui";</tag></item>
   </one-of>
  </rule>
  <rule id="non“>
   …
```

Comportement

Définit la règle racine
$reponse = oui | non

Définit la règle
$oui = oui | ouais | bien sûr

On remplace le mot reconnu par “oui”

Définit la règle
$non = …

Stochastic Language Models (N-Gram) Specification

- Définit :
  - Lexique
  - Évènements linguistiques
    - Comptes ngrammes
    - Structure arborescente
  - Eventuellement : facteurs de replis

- Permet:
  - Ngrammes distant
  - Interpolation de modèles
  - Classes grammaticales
  - Marqueurs sémantiques

Exemple

Écriture exhaustive

```
<lexicon>
 <token index="1"> A </token>
 <token index="2"> B </token>
 <token index="3"> C </token>
</lexicon>

<tree>
 <node branches="3" count="5" />
 <node index="1" branches="1" count="2" />
 <node index="2" branches="2" count="2" />
 <node index="1" count="1" />
 <node index="3" count="1" />
 <node index="2" branches="2" count="2" />
 <node index="1" branches="1" count="1" />
 <node index="2" count="1" />
 <node index="3" count="1" />
 <node index="3" count="1" />
</tree>
```

Écriture compacte

```
<lexicon>
 <token index="1"> A </token>
 <token index="2"> B </token>
 <token index="3"> C </token>
</lexicon>

<tree>
 3,5;
 1,1,2;
 2,2,2;
 1,1;
 3,1;
 2,2,2;
 1,1,1;
 2,1;
 3,1;
 3,1;
</tree>
```

Speech Synthesis Markup Language - SSML

- Lecture de messages pré-enregistrés
- Contrôle la synthèse vocale
  - Prononciation
  - Volume sonore
  - Hauteur tonale
  - Rythme
  - Genre
  - Voix particulière

Exemple

Fichier SSML

```
<?xml version="1.0"?>
 <speak version="1.0" xml:lang="fr-FR">
  <voice gender="female">
   Et maintenant, la palme du meilleur film pour l'année 2009
  </voice>
  <voice name="Michel">
    Cette année, la palme d'or est décernée à <break/>
    <emphasis level="strong">
      Michael Haneke
    </emphasis>
    <prosody rate="fast">
      pour son film Le Ruban Blanc
    </prosody>
  </voice>
 </speak>
```

Comportement

Prononce la phrase avec une voix de femme

Marque une pause
Prononce le nom avec une emphase importante

Prononce la fin de la phrase rapidement

Pronunciation Lexicon Specification - PLS

- Faire correspondre les termes et leur prononciation
  - Basé sur l'International Pronounciation Alphabet (IPA)
- Variantes de prononciation
  - Utilisable en reconnaissance

Exemple

```
<?xml version="1.0" encoding="UTF-8"?>
<lexicon version="1.0" xmlns="http://www.w3.org/2005/01/pronunciation-lexicon"
     alphabet="ipa" xml:lang="en-US">
  <lexeme>
    <grapheme>La vita è bella</grapheme>
    <phoneme>'lɑ 'vi:rə 'ʔeɪ 'bɛlə</phoneme>
    <!-- IPA string is:
     "&#x2C8;l&#x251; &#x2C8;vi&#x2D0;&#x27E;&#x259;
      &#x2C8;&#x294;e&#x26A; &#x2C8;b&#x25B;l&#x259;" -->
  </lexeme>
  <lexeme>
    <grapheme>Roberto</grapheme>
    <phoneme>ɹə'bɛ:ɹroʊ</phoneme>
    <!-- IPA string is:
     "&#x279;&#x259;&#x2C8;b&#x25B;&#x2D0;&#x279;&#x27E;o&#x28A;" -->
  </lexeme>
  <lexeme>
    <grapheme>Benigni</grapheme>
    <phoneme>bɛ'ni:nji</phoneme>
    <!-- IPA string is:
     "b&#x25B;&#x2C8;ni&#x2D0;nji" -->
  </lexeme>
</lexicon>
```

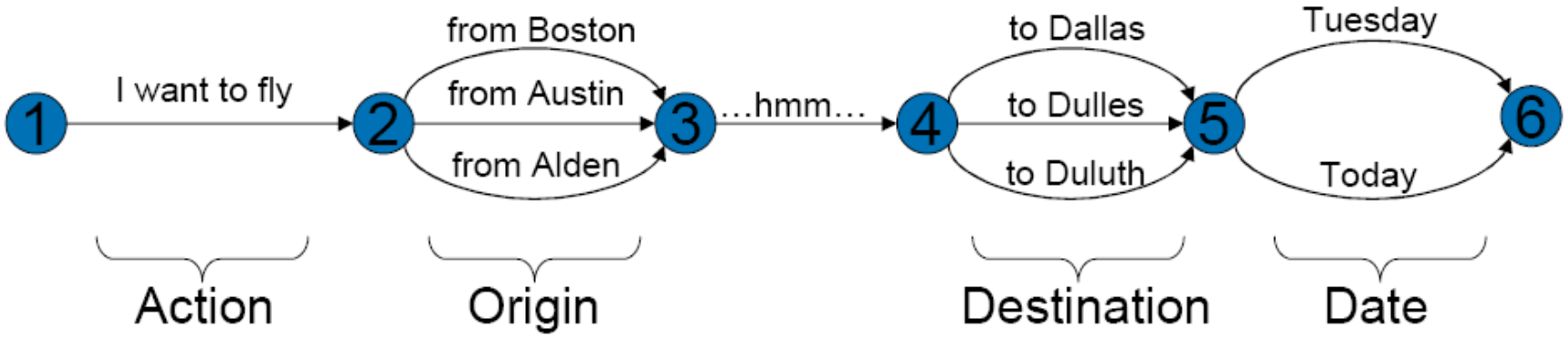
Semantic Interpretation for Speech Recognition - SISR

- Définit des annotations sémantiques associées à des règles de grammaire
- Mécanisme de substitution/transduction
- Permet une interprétation sémantique
  - ECMAScript
    - Traitements
    - Création d'objets complexes

Extensible Multi-Modal Annotation language - EMMA

- Standard issu du W3C Multimodal Interaction working group
- Standard pour décrire l'interprétation sémantique d'une entrée utilisateur
- Permet de définir un treillis d'interprétation

Exemple




```
<emma:lattice initial="1" final="6">
  <emma:arc from="1" to="2" emma:confidence="1.0"><action>airline</action></emma:arc>
  <emma:arc from="2" to="3" emma:confidence="0.7"><origin>BOS</origin></emma:arc>
  <emma:arc from="2" to="3" emma:confidence="0.5"><origin>AUS</origin></emma:arc>
  <emma:arc from="2" to="3" emma:confidence="0.1"><origin>IL05</origin></emma:arc>
  <emma:arc from="3" to="4"/>
  <emma:arc from="4" to="5" emma:confidence="0.6"><dest>DFW</dest></emma:arc>
  <emma:arc from="4" to="5" emma:confidence="0.2"><dest>IUD</dest></emma:arc>
  <emma:arc from="4" to="5" emma:confidence="0.1"><dest>DLH</dest></emma:arc>
  <emma:arc from="5" to="6" emma:confidence="0.4">
      <date><month>8</month><day>15</day><year>2006</year></date></emma:arc>
  <emma:arc from="5" to="6" emma:confidence="0.3">
      <date><month>8</month><day>10</day><year>2006</year></date></emma:arc>
</emma:lattice>
```

Call Control XML - CCXML

- Standard pour le traitement des appels
  - Orienté sessions d'appel
  - Gestion des évènements

- Fonctionnalités:
  - Routage et transfert d'appel
    - En fonction des ressources disponibles
    - Suivre un usager (localisations multiples)
    - Appeler un usager/service extérieur
  - Pontage
    - Mettre en relation deux interlocuteurs (au sein du système)

Call Control XML - CCXML (suite)

- Conférence
- Supervision
  - Ecoute d'une conversation (pas d'intervention possible)
- Réponse sélective
  - Décider de répondre ou non en fonction des informations d'appel
- Répondeur
- Dialogue
  - Utilisation d'une application VoiceXML
- Interaction avec d'autres serveurs

State Chart XML (SCXML)

State Machine Notation for Control Abstraction

- Basé sur les diagrammes d'états de D.Harel
  - Utilisés dans UML
  - Permet la représentation d'une machine à état
    - Etats
    - Transitions
    - Évènements
    - Conditions
  - Ajoute
    - Hiérarchisation
    - Parallélisation
    - Gestion des actions (E/S état, transition, état)

Compléments

Conception d'une application vocale
Les industriels du secteur
Les outils open-source

Conception d'une application vocale

- Définition du service
- Définition des interactions utilisateurs
  - Actions
  - Compréhension
    - Grammaire
- Définition des flux
  - Données
  - Appels
- Définition de la présentation
  - Messages

Les industriels du secteur

- Plateforme VoiceXML
  - Loquendo
  - Eloquant
  - Atos Worldwide
  - Voxeo
  - TellMe
  - Telisma
- Moteur de reconnaissance
  - Nuance
  - Voxeo
  - Lumenvox
  - Telisma

Les outils open-source

- Editeur VoiceXML
  - Java VoiceXML Editor
- Interpréteur VoiceXML
  - Java VoiceXML Interpreter
  - OpenVXI (Scansoft)
- CCXML
  - Voice Conference Manager
  - BladewareVXML

Les outils open-source (suite)

- Les systèmes de dialogue
  - Zanzibar Open IVR
  - Voxy (VoiceXML+Asterix)
  - Ravenclaw/Olympus (CMU/pas VoiceXML)

Perspectives

Perspectives

- VoiceXML 3.0 (2011)
  - Biométrie (identification / vérification)
  - File d'attente des interactions (SCXML)
    - Gestion du dialogue contrainte
  - Multimodalité
    - Synchronized Multimedia Integration Language (SMIL)
    - Exemple: Gestion de la vidéo
    - Diffusion bidirectionnelle : Lecture/Enregistrement

Perspectives (suite)

- Conception d'application vocales
  - Programmation de haut niveau (modèle)
  - Modularisation
  - Réutilisabité de composants
- Interaction avec les autres services web
  - Continuer avec le protocole HTTP 1.1 (1997/1999)?
- Sortie du cadre web/téléphonie

Perspectives (suite)

- Modèle descriptif (navigation)→ modèle cognitif (interaction/coopération)
  - Etats de dialogue →
- HumanML (OASIS)
  - Décrire les actes de communication
    - Physique
    - Kynésique
    - Culturel
    - Social
- EmotionML (W3C)
  - Annotation
  - Représentation

Perspectives (suite)

- Métrique d'évaluation

- Spoken Dialogue Challenge (orienté service)
  - Tâche : renseignements sur un service de bus
  - Mode opératoire : scenarii
  - Qualité de service

Merci pour votre attention

Références

- Contexte
  - http://www.w3.org/
- VoiceXML
  - http://www.w3.org/TR/voicexml21/
  - http://cafe.bevocal.com/docs/tutorial/index.html
- CCXML
  - http://www.voxeo.com/library/ccxml.jsp
  - http://www.developer.com/voice/article.php/11062_1565751_2/Introduction-to-CCXML-Part-I.htm
- Diagramme PLS
  - http://www-sop.inria.fr/acacia/personnel/Fabien.Gandon/lecture/Aristote_7Dec06_FabienGandon.ppt
- Diagramme EMMA
  - http://www.w3.org/Voice/2006/voicexml3.pdf
- Java VoiceXML Editor
  - http://www.speech.cs.cmu.edu/openvxi/index.html
- Java VoiceXML Interpreter
  - http://jvoicexml.sourceforge.net/
- OpenVXI
  - http://www.speech.cs.cmu.edu/openvxi/index.html
- Zanzibar Open IVR
  - http://www.spokentech.org/openivr/index.html
- Spoken Dialogue Challenge
  - http://www.signalprocessingsociety.org/technical-committees/list/sl-tc/spl-nl/2009-07/spoken-dialog-challenge/
  - http://dialrc.org/sdc/
- Olympus
  - http://wiki.speech.cs.cmu.edu/olympus/index.php/Olympus