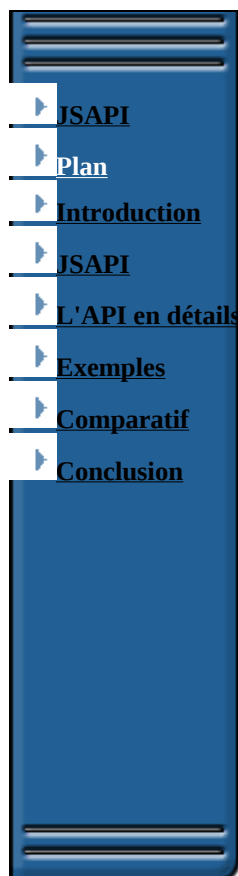


L'API Java Speech

JSAPI

L'API Java Speech - Xavier RODRIGUEZ



• Introduction

• JSAPI

L'architecture de l'API, les utilisations, l'architecture des applications

• L'API en détails

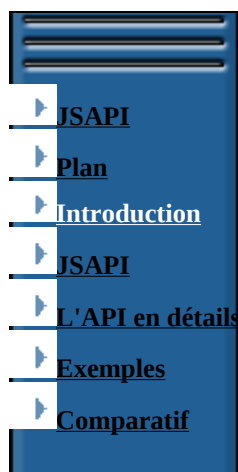
Les "classes" java, les grammaires

• Exemples

• Comparatifs

• Conclusion

L'API Java Speech - Xavier RODRIGUEZ



Java Speech

- Java Speech est une API permettant aux programmes écrits en JAVA d'utiliser la technologie de synthèse vocale et de reconnaissance vocale.
- SUN conjointement à Apple, AT&T, Dragon Systems, IBM, Novell, Philips et Texas Instruments

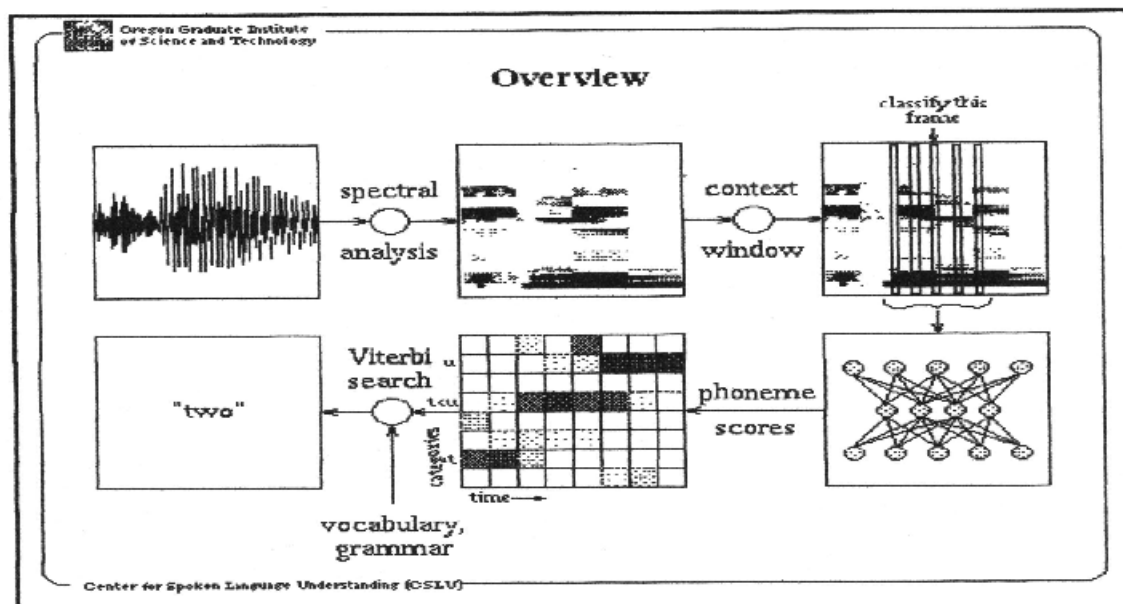
Conclusion

ont écrit les spécifications de cette API.

- Mise en service le 16 octobre 1998.

L'API Java Speech - Xavier RODRIGUEZ

Reconnaissance vocale



L'API Java Speech - Xavier RODRIGUEZ

Synthèse vocale

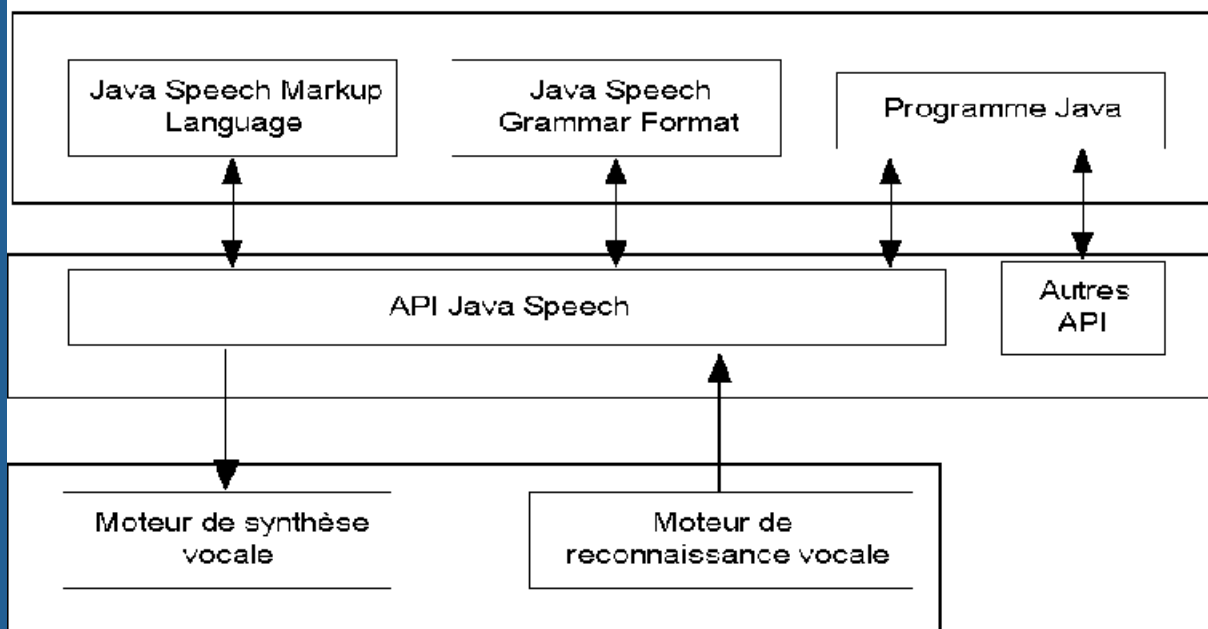
Consiste à convertir du texte écrit en phrases parlés

La synthèse s'effectue en plusieurs temps :

- L'analyse du texte en paragraphes et en phrases pour une meilleure intonation
- La détection de constructions idiomatiques (abréviations, lecture des dates, des acronymes, ...)
- La conversion de chaque mot en phonèmes
- Traiter la prosodie (rythme), la mélodie de l'élocution, l'emphase sur certains mots de la phrase, ...
- La génération des sons correspondants

L'API Java Speech - Xavier RODRIGUEZ

Architecture de JSAPI



L'API Java Speech - Xavier RODRIGUEZ

Les utilisations

• Applications bureautiques

Dictée vocale, l'utilisateur de l'application a les mains occupées pendant l'utilisation de l'application, applications pour personnes ayant un handicap, ...

• Applications téléphoniques

Consultation de messages, applications nécessitant l'utilisation de caractères, standard téléphonique, ...

• Systèmes embarqués

Systèmes sans clavier / écran, automates, ...

• Internet

L'API Java Speech - Xavier RODRIGUEZ

Quand utiliser JSAPI?

JSAPI
L'API en détails
Exemples
Comparatif
Conclusion

A utiliser	A éviter
• L'activité de l'utilisateur l'oblige à regarder quelque chose d'autre que l'écran • Le dispositif ne contient pas d'écran • L'interface utilisateur doit incarné une personne (borne interactive) • Handicap de l'utilisateur	• La quantité d'information à présenter est trop importante • L'utilisateur doit comparer plusieurs données • Les informations à présenter sont confidentielles
• Le clavier n'est pas disponible • Les mains de l'utilisateur sont occupées par une autre tâche. • L'utilisateur ne peut pas ou ne veut pas saisir au clavier • Handicap de l'utilisateur	• L'utilisateur doit parler à d'autres personnes pendant l'utilisation de l'application • L'utilisateur travaille dans un environnement bruyant • Les actions de l'application peuvent être réalisées de façon aisée avec un clavier et/ou une souris

L'API Java Speech - Xavier RODRIGUEZ

JSAPI
Plan
Introduction
JSAPI
L'API en détails
Exemples
Comparatif
Conclusion

Les différents moteurs de synthèse/reconnaissance vocale

- IBM ViaVoice (speech for java) sous Windows, Linux et Mac supporte 7 langues.
- Lernout & Hauspie's TTS sous Sun Solaris permet de gérer 10 voix différentes
- Conversa Wab sous Windows basé sur un navigateur Web
- Festival sous Unix, Sun Solaris et Windows supporte 3 langues

L'API Java Speech - Xavier RODRIGUEZ

JSAPI
Plan
Introduction
JSAPI
L'API en détails
Exemples
Comparatif

Les grammaires

- Java Speech Markup Language (JSML)

JSML est un langage basé sur langage XML

JSML permet d'indiquer la façon dont le texte doit être interprété par le synthétiseur.

Conclusion

• Java Speech Grammar Format (JSGF)

JSGF indique à l'ordinateur quels mots et dans quel ordre il doit les attendre de la part de l'utilisateur

Exemple:

```
RuleGrammar SimpleCommand;
public quitter= fermer | quitter;
```

L'API Java Speech - Xavier RODRIGUEZ

Le paquetage javax.speech

- Classes de javax.speech
- Classes de javax.speech.recognition
- Classes de javax.speech.synthesis

L'API Java Speech - Xavier RODRIGUEZ

Exemple de synthèse vocale

```
import javax.speech.*;
import javax.speech.synthesis.*;
public class Bienvenue {
    public static void main(String args[]) {
        String texteASynthesiser;
        texteASynthesiser="Bonjour";
        texteASynthesiser+="<EMP LEVEL=\"strong\">" + "mon nom est " + "</EMP>";

        texteASynthesiser+="<EMP LEVEL=\"reduced\">" + "Ordinateur" + "</EMP>";

        try {
            Synthesizer synth = Central.createSynthesizer( new
            SynthesizerModeDesc(Locale.FRENCH));
            synth.allocate();
            synth.speak(texteASynthesiser, null);
            synth.speak("fin du message", null);
            synth.deallocate();
        } catch (Exemption e) {
            e.printStackTrace();
        }
    }
}
```

L'API Java Speech - Xavier RODRIGUEZ

Exemple de reconnaissance vocale

```
import javax.speech.recognition.* ;
import java.io.FileReader ;
import java.util.Locale ;
Public class HelloWorld extends GrammarResultAdapter{
    static HelloWorld hello=new HelloWorld() ;
    public void acceptedResult(ResultEvent e){
        System.out.println("L utilisateur a dit bonjour");
        System.exit(0);
    }
public static void main(String args[]){
    try{
        RecognizerModeDesc require=new Recognizer(Locale.FRENCH);
        Recognizer rec=Central.createRecognizer(require);
        Rec.allocate();
        FileReader reader=new FileReader(args[0]);
        RuleGrammar gram=rec.loadJSGMF(reader);
        Gram.addGrammarListener(hello);
        Gram.setActive(true);
        Rec.commitChanges();
        Rec.resume();
    }catch (Exception e) {
        e.printStackTrace();
    }
}
```

L'API Java Speech - Xavier RODRIGUEZ

Voice XML ou Java Speech API ?

- **Avantages de VoiceXML**

- Basé sur le langage XML
- Facile à utiliser
- Incorpore la reconnaissance DTMF
- Incorpore de nombreuses fonctions téléphoniques (transfert d'appels, deconnection,...)

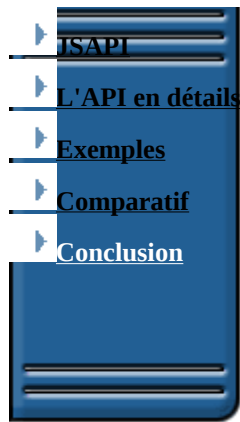
- **Inconvénients de VoiceXML**

- Nécessite l'utilisation d'une passerelle VoiceXML dans le cas d'une application téléphonique ou d'un interpréteur VoiceXML dans les autres cas

L'API Java Speech - Xavier RODRIGUEZ

Conclusion

Malgré un développement qui commence déjà à s'estomper à cause de l'apparition de VoiceXML, JSAPI



reste un bon moyen pour réaliser des applications vocales avec Java.

L'avenir sera marqué par l'apparition de nouveau moteurs de reconnaissance vocale combinant l'analyse sonore mais également l'analyse visuelle.

L'API Java Speech - Xavier RODRIGUEZ

This document was created with Win2PDF available at <http://www.daneprairie.com>.
The unregistered version of Win2PDF is for evaluation or non-commercial use only.