



Internet 新技术丛书

VoiceXML

语音应用程序开发

*Voice Application
Development with
VoiceXML*

(美) Rick Beasley 等著

王建华 王卫峰 等译



机械工业出版社
China Machine Press

SAMS

Internet新技术丛书

VoiceXML语音应用 程序开发

(美) Rick Beasley 等著

王建华 王卫峰 等译



机械工业出版社
China Machine Press

VoiceXML是一种基于XML的因特网标记语言,可以用来开发语音用户界面,使人们可以通过移动电话来访问因特网。本书通过VoiceXML的基本概念、VoiceXML的用户界面、VoiceXML应用程序的开发过程、VoiceXML的基本元素和属性以及VoiceXML的高级开发技术等专题,对VoiceXML应用程序的开发进行了详细的介绍。

本书内容翔实、深入浅出、极具参考价值,既适合VoiceXML应用程序的开发人员使用,也适合任何希望了解新型因特网访问技术的人员阅读。

Rick Beasley, et al: Voice Application Development with VoiceXML.

Authorized translation from the English language edition published by Sams, an imprint of Macmillan Computer Publishing U.S.A.

Copyright © 2001 by Sams. All rights reserved.

Chinese simplified language edition published by China Machine Press.

Copyright © 2002 by China Machine Press.

本书中文简体字版由美国麦克米兰公司授权机械工业出版社独家出版。未经出版者书面许可,不得以任何方式复制或抄袭本书内容。

版权所有,侵权必究。

本书版权登记号: 图字: 01-2001-4286

图书在版编目(CIP)数据

VoiceXML语音应用程序开发 / (美) 贝斯利(Beasley, R.)等著; 王建华等译. - 北京: 机械工业出版社, 2002.2

(Internet新技术丛书)

书名原文: Voice Application Development with VoiceXML

ISBN 7-111-09777-7

I. V… II. ① 贝… ② 王… III. 可扩充语言, XML - 程序设计 IV. TP312

中国版本图书馆CIP数据核字(2002)第001151号

机械工业出版社(北京市西城区百万庄大街22号 邮政编码 100037)

责任编辑: 梁开莉 张鸿斌

北京忠信诚胶印厂印刷·新华书店北京发行所发行

2002年2月第1版第1次印刷

787mm×1092mm 1/16·18印张

印数: 0 001-4 000册

定价: 35.00元

凡购本书, 如有倒页、脱页、缺页, 由本社发行部调换

译者序

近年来,移动通信技术得到了空前迅速的发展,移动电话的用户出现了爆炸式的增长。有人估计,到2005年,全球将有10亿部手机投入使用。由此人们对使用移动电话在任何时间和任何地点访问因特网产生了浓厚的兴趣,并且这已经成为许多人的普遍愿望。可以预言,届时,利用语音技术通过移动电话来访问因特网将会成为一种非常流行的手段。

VoiceXML(语音扩展标记语言)是一种基于XML(扩展标记语言)的因特网标记语言;它可以用来开发语音用户界面。通过语音用户界面,用户就可以通过电话来访问因特网上的内容,而不是通过计算机键盘和显示器与应用程序进行交互操作。为此我们可以将VoiceXML视为用于电话的HTML语言。

本书是专门为了满足从事VoiceXML应用程序开发的广大编程人员的需要而编写的。全书分几个专题对VoiceXML应用程序的开发进行了全面的介绍。第1~3章介绍VoiceXML的基本概念,讲述你开始学习VoiceXML时首先需要掌握的基本知识。第4~6章介绍VoiceXML用户界面(VUI),说明与VoiceXML的用户界面有关的各种问题。第7章介绍VoiceXML应用程序的开发过程以及需要考虑的有关问题。第8、9章介绍VoiceXML应用程序的开发基础,详细讲述了VoiceXML的基本元素和属性。第10~12章介绍VoiceXML的高级开发技术,分别讲述支持访问特定平台的函数、ECMAScript的基本概念、ECMAScript与VoiceXML的集成、进行语音录制和电话呼叫转接时使用的各种元素,以及用VoiceXML进行服务器端编程的方法。第13章对涉及VoiceXML的未来发展的某些问题进行了展望。

本书配有3个附录。附录A提供了一个VoiceXML 1.0版的元素列表,对每个元素的功能进行了简要的介绍。附录B给出了本书开发的完整的在线书店应用程序。附录C简要介绍了VoiceXML 2.0版对1.0版可能进行的修改。

本书既适合从事VoiceXML应用程序开发的软件技术人员使用,也适合任何需要或希望了解新型因特网访问技术的人员阅读参考。

本书由王建华、王卫峰、杨惠琼、叶兆民、张晓佳等翻译,席赛珠校对,王卫峰、芳妮录入。由于译者水平所限,不妥之处敬请读者指正。

2001年11月

AMJS07/0706

前 言

“到2005年时，随着10亿部手机投入使用，语音肯定会成为你离开办公桌后访问因特网时所使用的最流行的（尽管不是最普及的）手段。VoiceXML是一种将电话与电信运营商、服务提供商、开发商和消费者联系在一起的工具。”（引自Minde公司的Web站点，网址是<http://www.minde.com/solutions/vxml.html>。）

目前，用移动电话访问因特网看来已经成为每个人的普遍愿望。在任何时间、任何地点访问因特网的梦想，使人们对此产生了浓厚的兴趣，并且希望尝试使用如VoiceXML和WML（无线标记语言）等瘦客户移动支持技术来访问因特网。尽管在美国很少采用像WML那样的小型屏幕浏览器访问技术，但是，由于人们始终希望采用移动式因特网访问技术，因此人们对VoiceXML和语音门户技术产生了更加浓厚的兴趣。

人们对以非PC为基础的移动电话访问因特网的技术的兴趣始终未减。如果你正在开发因特网应用程序或服务，那么你将很快有机会使用瘦客户机访问方式(如果你还没有使用的话)。

如果你已经在开发因特网服务程序，可能会想到VoiceXML简直是“妙极了！它完全是另一种*ML（标记语言）”。虽然VoiceXML是一种比较易学的标记语言，但你很快会发现语音应用程序具有许多开发人员前所未遇的复杂性。本书将有助于你学习如何解决语音应用程序开发中你所不熟悉的一些问题，包括它们的支持结构和使用的平台、开发过程的不同、VoiceXML的元素和属性、语法和VoiceXML的用户界面。

本书的读者对象

本书编写的目的是为了满足不同从事或涉及因特网相关技术开发的人们进行学习的需要。这些人包括：

- 为VoiceXML应用程序进行结构设计或制定技术要求的工程师。
- 从事VoiceXML应用程序开发的软件开发人员。
- 为VoiceXML应用程序制定和执行测试方案的软件测试人员。
- 负责设计VoiceXML应用程序的提示、消息或帮助信息的人类行为工程学家或其他专家。
- 需要了解VoiceXML应用程序的功能和结构的销售人员和市场专业人员。
- 当因特网服务提供商开始托管VoiceXML应用程序时，那些希望首先利用这些应用程序的业余爱好者。
- 任何需要或希望了解重要的新型因特网访问技术的人。

要想从本书中获得最大收益需要具备哪些条件

若要从本书中获取最大的收益，你需要具备的条件如下：

- VoiceXML的说明书，这是你了解VoiceXML元素和属性信息的最佳参考资料，你可从VoiceXML论坛的Web站点上获得该说明书，它的网址是<http://www.voicexml.org/>。
- 能够运行你希望使用的任何专用软件开发工具包（SDK）的计算机平台（参见软件开发工具包供应商的说明书），它能够提供基本的因特网访问工具，并且可以使用文本编辑器或者XML编辑器来编辑VoiceXML文件。
- 能够访问模拟器、测试环境或语音浏览器平台。有关VoiceXML工具和对开发人员的支持特性的详细信息请参见第3章。
- 具备电话接入功能，这样你就可以通过拨号呼叫对环境进行测试或者建立环境的原型。

本书中的示例代码是使用的IBM公司的WebSphere Voice Server SDK（语音服务器软件开发工具包）进行测试的，若要获得该SDK，请访问网址<http://www-4.ibm.com/software/speech/enterprise/ep-12.html>（除了呼叫转移和录音以外的全部功能）。另外，你还可以从VoiceGenic开发人员的工作室站点获得，网址是<http://developer.voicegenie.com/>（包括呼叫转移和录音）。

本书的结构

本书是按照以下几个专题进行编排的：

- VoiceXML的基本概念——本书第1至3章介绍了开始学习VoiceXML时首先需要掌握的有关信息。第1章讲述了VoiceXML的发展历史以及它在因特网应用程序中的使用情况。第2章介绍用于提供电话访问VoiceXML应用程序所用的结构。第3章详细解释了一些可以利用的软件工具和环境。
- VoiceXML用户界面（VUI）——第4至6章介绍了与VoiceXML的用户界面有关的各种问题。第4章介绍用户界面和VoiceXML用户界面的基本概念。第5章从以用户为中心的设计进程的角度来说明VUI的设计。第6章介绍建造原型、评估以及测试应用程序可用性的方法。
- 开发过程——第7章介绍VoiceXML应用程序的开发过程以及需要考虑的有关问题。
- VoiceXML应用程序的开发基础——第8章和第9章介绍着手进行VoiceXML应用程序开发时所需要的基本元素。第8章详细介绍VoiceXML的基本结构、变量和定位技术。第9章介绍与接收用户输入和传递应用程序输出相关的元素和技术。
- VoiceXML的高级开发技术——第10、11、12章以高级VoiceXML应用程序开发为主题。第10章介绍支持访问特定平台的函数、将ECMAScript与VoiceXML相集成、进行语音录制和电话呼叫转接时使用的各种元素。第11章介绍ECMAScript的基本概念。第12章介绍用VoiceXML进行服务器端编程的方法。
- 预测——第13章讲述了随着VoiceXML的持续发展，对某些问题的展望。
- 附录——附录A提供了一个VoiceXML 1.0版的元素列表、每个元素的简要介绍与介绍它的章节。附录B给出了本书开发的完整的在线书店应用程序。附录C简要地介绍了一旦VoiceXML 2.0版推向市场，它对1.0版可能进行的修改。

作者简介

Rick Beasley是朗讯公司的执行经理，他目前管理着几个软件开发项目。他的小组负责开发综合性的因特网/电信服务程序和平台，其中包括Milife Voice Solution这个VoiceXML网关。

在过去的3年里，Rick为移动因特网开发的软件既有网络基础软件也有用户应用程序。他在12年的软件开发工作中开发的产品包括网络管理软件、手持平台和地理信息系统。Rick目前正在研究“蓝牙”（Blue tooth）和基于位置的服务程序。同时，Rick还是与VoiceXML服务程序有关的两项专利的合伙申请人。

Mike Farley 是朗讯公司的一名专业技术人员。目前负责从事以用户为中心的软件和信息产品设计。

Mike在无线和有线电信系统及智能网络方面拥有35年的客户服务经验和面向用户的工作经验。他为通过用户为本的设计、方便的交叉功能设计和开发小组、技术培训以及对进程和组织的再设计，为用户提供了人性化的支持。

在过去的两年中，Mike负责为实现VoiceXML、无线标记语言（WML）、Web浏览器和PC用户界面的各式各样的服务应用程序进行以用户为中心的设计。

John O'Reilly是朗讯公司的一名专业技术人员。目前负责领导朗讯公司的VoiceXML平台小组的工作。从1999年3月起，John一方面参与VoiceXML平台小组的工作，一方面又担任VoiceXML应用程序开发小组的主要开发人员。John作为一名专题发言人，在2001冬季因特网之声（VON）上介绍了VoiceXML应用程序的开发问题。John也是一名与VoiceXML相关专利的合伙申请人。

Leon Henry Squire是朗讯公司一名专业技术人员。他领导的开发小组负责开发在增强型媒体资源服务器（eMRS）上运行的TelePortal VoiceXML引擎。Leon在AT&T和朗讯公司中从事了13年的电话技术的开发研究。在软件开发、系统测试和系统设计方面担当技术负责人的角色。他目前是朗讯公司的高级软件开发中心的主要软件开发人员，负责制定开发小组的技术发展方向，该中心位于北卡罗来纳州的卡里市。

目 录

译者序	
前言	
作者简介	
第1章 VoiceXML的概述	1
1.1 VoiceXML语言	1
1.2 一个VoiceXML应用程序方案	1
1.3 语音浏览	2
1.4 VoiceXML在语音浏览器中是怎样工作的	3
1.4.1 应用程序的输入	3
1.4.2 应用程序的输出	4
1.5 高级网络结构	4
1.6 VoiceXML网关	5
1.7 托管VoiceXML应用程序	6
1.8 选择VoiceXML应用程序	7
1.9 用VoiceXML来开发应用程序	7
1.10 支持的语言和工具	9
第2章 运行VoiceXML应用程序的网络	10
2.1 VoiceXML网关	10
2.1.1 TTS资源	11
2.1.2 ASR资源	12
2.1.3 VoiceXML浏览器和翻译器	13
2.1.4 声音资源软件	14
2.1.5 TCP/IP资源	14
2.1.6 电话资源	14
2.2 电话网络的拓扑	15
2.2.1 有线电话网的拓扑	15
2.2.2 无线电话网的拓扑	17
2.3 小结	19
第3章 开发人员使用的工具	20
3.1 开发环境的类型	20
3.2 可供选择的开发环境和它们的利弊	20
3.2.1 托管的开发环境	22
3.2.2 模拟的开发环境	27
3.3 基于Web的开发环境	28
3.3.1 BeVocal Café	29
3.3.2 HeyAnita FreeSpeech	30
3.3.3 Tellme Studio	31
3.3.4 VoiceGenie 的Developer Workshop	32
3.3.5 Voxeo Community	33
3.4 其他有用的工具	33
3.5 小结	34
第4章 VoiceXML用户界面	35
4.1 用户界面的基本概念	35
4.1.1 用户界面的组成部分	36
4.1.2 CHUI、GUI和WUI	36
4.1.3 Web用户界面	37
4.2 VUI概述	37
4.2.1 VUI的功能	37
4.2.2 VUI的术语	38
4.2.3 VUI的输入/输出	38
4.3 VUI与GUI在设计上的差异	40
4.3.1 Web浏览器用户界面与VUI的比较	40
4.3.2 Web GUI与VUI的比较总结	42
4.4 VUI用户的特点	43
4.4.1 对PC和因特网的使用经验有限	43
4.4.2 处于移动环境之中	44
4.4.3 单一的输入/输出模式	44
4.5 设计VUI时应该考虑的其他重要问题	44
4.5.1 吸引你的目标用户	45
4.5.2 能够代表目标用户群希望执行的任务	45
4.5.3 提供一个适合代码开发的结构	45
4.5.4 尽量减少限制条件造成的影响	46

4.6 小结	46	6.5.1 可用性测试的前提条件	74
第5章 VUI设计的概述	47	6.5.2 可用性测试过程的概述	74
5.1 软件开发过程的概述	47	6.5.3 制定可用性测试方案	75
5.2 UCD过程的概述	48	6.5.4 使用测试结果来改进应用程序 的设计	77
5.3 可用性测试	52	6.5.5 完成可用性测试的工作周期	78
5.3.1 VoiceXML的可用性测试	52	6.6 小结	78
5.3.2 测试原则1: 尽量降低对用户的 记忆和识别过程的要求	53	第7章 VoiceXML应用程序的开发过程	79
5.3.3 测试原则2: 保持简单而自然的对话框	55	7.1 确定应用程序开发的要求	79
5.3.4 测试原则3: 提供清楚的导航和 快捷操作方式	56	7.1.1 应用程序部署地区使用的口语	79
5.3.5 测试原则4: 提供一致的反馈信息 和便于使用的错误处理工具	57	7.1.2 语法的定义	80
5.3.6 测试原则5: 保持导航和术语的一致性	59	7.1.3 语音识别与TTS软件工具的选择	80
5.3.7 测试原则6: 对选择的语音和声音 的个性进行评估	59	7.1.4 支持的VoiceXML应用程序技术规范	80
5.4 UCD和系统设计的参考资料	61	7.1.5 进行数据上的准备	80
5.5 小结	61	7.1.6 内容提供商界面	81
第6章 VUI应用程序原型和可用性测试	62	7.1.7 性能、容量和可靠性	81
6.1 UCD的前端工作	62	7.2 确定应用程序的结构	82
6.2 应用程序的任务分析	64	7.3 应用程序设计的实现方法	83
6.2.1 新技术应用程序开发中的任务分析	64	7.4 应用程序性能计量参数的定义	83
6.2.2 任务分析的举例	64	7.5 VoiceXML应用程序的测试	85
6.3 早期的应用程序原型——应用程序 流程图	65	7.6 应用程序的部署	85
6.3.1 建立低保真度的应用程序原型	66	7.7 小结	86
6.3.2 用流程图来说明应用程序的功能	67	第8章 VoiceXML的基本命令	87
6.3.3 根据流程图对应用程序的可用性 进行评估	67	8.1 第一个VoiceXML程序	87
6.3.4 应用程序可用性要求举例	68	8.2 xml元素	87
6.3.5 初始语法设计	69	8.3 注释	87
6.3.6 与开发人员交流信息的结果	71	8.4 vxml元素	88
6.4 VUI脚本——中间阶段的应用程序原型	71	8.5 form元素	88
6.4.1 可用性测试举例	71	8.6 窗体中的项目	88
6.4.2 可重复使用的资源	73	8.6.1 name属性	88
6.4.3 定时问题与资源要求	73	8.6.2 expr属性	88
6.5 VoiceXML代码原型	74	8.6.3 cond属性	88
		8.6.4 使用说明	89
		8.6.5 block元素	89
		8.6.6 name属性	89
		8.6.7 expr属性	89
		8.6.8 cond属性	90

8.6.9 使用说明	90	8.12.6 namelist 属性	104
8.6.10 字段项	90	8.12.7 method 属性	104
8.6.11 filled元素	90	8.12.8 enctype属性	104
8.7 meta元素	90	8.12.9 caching 属性	104
8.7.1 name属性	91	8.12.10 fetchtimeout 属性	104
8.7.2 content属性	91	8.12.11 fetchhint 属性	104
8.7.3 http-equiv属性	91	8.12.12 fetchaudio属性	104
8.8 变量	92	8.12.13 return 元素	105
8.8.1 声明变量	93	8.12.14 程序举例	105
8.8.2 name属性	93	8.12.15 另一个程序例子	106
8.8.3 expr 属性	93	8.13 goto 元素	108
8.8.4 使用说明	93	8.13.1 next属性	108
8.8.5 name属性	94	8.13.2 expr属性	109
8.8.6 expr属性	94	8.13.3 nextitem属性	109
8.8.7 使用说明	94	8.13.4 expritem属性	109
8.8.8 关于变量的其他信息	94	8.13.5 caching属性	109
8.8.9 根文档	95	8.13.6 fetchtimeout 属性	109
8.8.10 会话变量	97	8.13.7 fetchhint 属性	109
8.8.11 影子变量	97	8.13.8 fetchaudio属性	109
8.8.12 变量命名冲突	97	8.13.9 程序举例	109
8.9 if元素	98	8.14 clear元素	111
8.10 if / elseif / else元素	99	8.14.1 namelist属性	111
8.10.1 cond 属性	100	8.14.2 使用说明	111
8.10.2 程序举例	100	8.15 循环	111
8.10.3 嵌套的if/else/elseif 语句	101	8.16 exit元素	112
8.11 param元素	102	8.16.1 expr属性	112
8.11.1 name属性	102	8.16.2 namelist属性	112
8.11.2 expr 属性	102	8.16.3 程序举例	113
8.11.3 value 属性	103	8.17 disconnect元素	113
8.11.4 valuetype属性	103	8.18 小结	114
8.11.5 type属性	103	第9章 VoiceXML的输入和输出命令	115
8.12 subdialog元素	103	9.1 VoiceXML中的输出方法	115
8.12.1 name 属性	103	9.1.1 prompt 元素	116
8.12.2 src属性	103	9.1.2 语音标记	118
8.12.3 expr 属性	103	9.1.3 value元素	122
8.12.4 cond 属性	104	9.1.4 程序举例	122
8.12.5 modal属性	104	9.1.5 audio元素	126

9.1.6 使用举例	126	11.4.2 Date对象	213
9.2 接收用户的输入	131	11.4.3 Math 对象	214
9.2.1 field 元素	131	11.4.4 String对象	215
9.2.2 option元素	140	11.4.5 函数与 Function对象	215
9.2.3 enumerate元素	147	11.5 运算符	216
9.2.4 menu元素	147	11.6 类型转换	218
9.2.5 choice元素	150	11.7 控制程序的执行	218
9.2.6 link元素	152	11.7.1 if/else语句	218
9.2.7 grammar元素	156	11.7.2 case 语句	219
9.2.8 dtmf元素	157	11.7.3 循环	220
9.3 事件	159	11.7.4 with语句	222
9.3.1 事件的类型	159	11.8 小结	222
9.3.2 默认的 catch元素	160	第12章 高级主题和服务端的应用	
9.3.3 catch元素	161	程序开发	223
9.3.4 throw 元素	168	12.1 语言	223
9.3.5 error元素、help元素、noinput		12.2 创建VoiceXML应用程序	224
元素和nomatch元素	171	12.2.1 Java小服务程序	224
9.3.6 reprompt元素	172	12.2.2 PHP脚本程序	229
9.3.7 initial 元素	175	12.2.3 数据库访问	231
9.4 小结	178	12.3 加载声音文件	239
第10章 VoiceXML的高级元素	179	12.4 小结	245
10.1 record元素	179	第13章 VoiceXML的未来发展前景	246
10.2 transfer元素	180	13.1 关于语音合成标记语言的概述	246
10.3 script 元素	182	13.2 关于语音识别语法的概述	247
10.4 property元素	185	13.3 多模式应用程序	248
10.5 object元素	196	13.3.1 多模式应用程序给VUI带来的影响	249
10.6 submit元素	198	13.3.2 串行输入/输出	249
10.7 完成应用程序中VoiceXML部分的开发	201	13.3.3 非协调式同时输入/输出	250
10.8 小结	209	13.3.4 协调式同时输入/输出	250
第11章 VoiceXML中的ECMAScript	210	13.4 未来用户界面的复杂性	251
11.1 ECMAScript注释	210	13.5 VoiceXML多模应用程序的现状	252
11.2 变量的类型	210	13.6 小结	252
11.3 用户定义的对象	211	附录A VoiceXML 1.0的元素	253
11.4 内置对象	212	附录B 在线书店应用程序	255
11.4.1 Array 对象	213	附录C VoiceXML 2.0的有关情况	277

第1章 VoiceXML的概述

1.1 VoiceXML语言

VoiceXML（语音扩展标记语言）是一种基于XML的因特网标记语言，用于开发语音用户界面。它是“语音Web”使用的语言，它使得用户可以使用电话来访问因特网托管的内容。我们可以将其视为用于电话的HTML。

使用语音界面，用户可以用他们的声音而不是通过计算机键盘和显示器与应用程序进行交互操作。在银行、学校或证券交易所，你可能已经通过语音邮件或者交互式语音应答（IVR）系统接触过语音界面。这些系统提示用户输入相应的信息，用户再根据要求输入相应的数据作为应答，然后系统就可以为用户执行有关的在线操作，例如，在不同账户之间进行资金转账。

在语音界面中，有一个VoiceXML对话框（大致相当于一个HTML页）用于描述应用程序对用户说出的提示信息，定义和接收来自用户的应答，并且描述程序控制的流程。另外，VoiceXML还给电话规定了一些特殊的操作，例如，从一个VoiceXML应用程序转移到另一个电话呼叫的功能。

用户通过拨打应用程序的电话号码来访问VoiceXML应用程序。从用户的角度看，这个电话号码相当于一个Web网页的统一资源地址（URL）。用户可以通过任何类型的电话拨打，包括陆路电话、移动电话和卫星电话。

电话的类型将会改变语音通过电话网络的传输方式，但对于大多数应用程序开发人员来说，这并没有什么关系。我们将在第2章“运行VoiceXML的网络”中讲述这个问题。

为什么要创造一种新语言？

早在着手开发语音应用程序之前，VoiceXML的设计者就决定创造一种新的标记语言，而不是扩充HTML的功能。这是因为采用语音的操作界面与Web浏览器一类的视觉显示方式有着本质的区别。

1.2 一个VoiceXML应用程序方案

要弄懂VoiceXML是怎样运行的，最简单的方法就是从头到尾将一个小用户应用程序方案运行一遍。下面是一个简单的例子，我们将在全书中引用它。在我们的应用程序方案中，下面这个对话是在一个呼叫者与一个图书销售商的服务程序之间进行的：

服务器程序：欢迎光临Acme Books在线书店，你可以在任何时候说“Top Menu”（主菜单），以便返回本菜单。说“Info”，就可以听到我们书店的营业时间。说“Check out”（结账）就可以完成购买。说出书的名字，就可将书放入购物车中。

用户：Info。

服务程序：Acme Books在线书店全年365天、全天24小时营业。

用户：再见。

服务程序：感谢你访问Acme Books在线书店。

下面是在更高层次上该在线书店的服务程序与用户进行谈话时必须采取的操作步骤：

1) 用户拨打在线书店服务程序的电话号码。

2) 负责托管该在线书店服务的语音服务提供商（VSP）的VoiceXML网关，接收到这个电话呼叫和该呼叫的有关信息。例如，被呼叫的号码（用户拨打的号码）和主叫号码（用户的电话号码）。

3) VoiceXML网关搜索一个数据库，并且将被呼叫的电话号码转换为一个统一资源地址（URL），这个URL就是Web服务器上的在线书店服务程序的VoiceXML主页（books.vxml）的位置。

4) VoiceXML网关激活books.vxml主页，如果预先进行了设定，那么也可以激活相关的文件，例如，语法文件或录制的声音文件。相关的文件可以被高速缓存在VoiceXML网关上。

5) VoiceXML网关对VoiceXML进行翻译，一步一步执行books.vxml文件，并根据应用程序的规定与用户之间进行交互操作。

6) 根据需要，从Web服务器上下载其他的VoiceXML文件和相关文件。

1.3 语音浏览

根据World Wide Web联盟的“语音浏览器的活动”的报告，“语音浏览器允许每个人通过任何电话来享受基于Web的服务，从而使人们可以在任何时间和任何地点，无论是在家中，在路上，还是在办公室，都可以访问Web。”

语音浏览器是驻留在网络上的一种应用程序，它位于语音、电话和因特网组件之上，负责将VoiceXML应用程序传递给用户。语音浏览器由负责解析VoiceXML文档的VoiceXML翻译器以及连接语音和电话组件的界面组成。

由于语音浏览器的概念与HTML或无线通信技术相关，而浏览器又位于终端设备上，因此位于网络上的语音浏览器的概念似乎有点儿让人搞不清楚。在这里，语音浏览器与HTML和WAP（无线应用协议）很难进行比较。在下一节中，当我们把语音浏览器放在网络环境中来加以说明时，语音浏览器的功能就会非常明显地体现出来。我们还将介绍VoiceXML网关的功能，它是托管语音浏览器的实际网络组件。而现在则让我们将注意力集中在浏览器的功能——浏览上。

语音浏览器的巨大潜力存在于无所不在的电话和独特的语音应用中。对于世界上的许多人来说，也许他们无法做到人人拥有一台PC机，但是在发达国家和发展中国家中，电话通常都被视为一种标准的家用电器设备。

在某些情况下，例如用户正在驾驶汽车时，如果他想要访问因特网，那么他只有使用免提电话和像语音那样的免提界面，才能非常安全和舒适地上网。例如，通过语音浏览器来查询交通路况信息，并不比你一边开车一边拨打任何其他电话更困难。相比之下，利用无线设备的图形浏览器来查询因特网的内容，则需要进行一定的身体和脑力上的训练，并且费用相当昂贵。

带有某些残疾的用户也可以很好地利用语音服务系统。由于没有考虑到用户的能力，有些

用HTML和WML开发出来的因特网站点和应用程序在使用时很不方便,因为这些服务更加需要正常的语言功能才能使用。这些内容一旦通过语音浏览器来显示,就可能吸引更广泛的听众。

许多内容提供商除了使用HTML和WML外,还用VoiceXML来传递他们的内容,目的是为了寻求最大数量的用户,就像他们匆匆忙忙地将无线服务添加到他们的传统因特网内容上去一样。

VoiceXML 的历史和演变发展

VoiceXML的起源可以追溯到1995年,当时AT&T公司研究部的计算机科学家开始着手Phone Web项目的开发。当朗讯公司脱离AT&T时,AT&T公司继续进行Phone Web项目的开发。这两家公司开发出不同版本的电话标记语言(PML)。最初的研究人员中有两个人进入了摩托罗拉公司,他们开发了VoxML。IBM公司也展开了类似的研究工作,从而导致SpeechML语言的问世。

这些公司成了VoiceXML论坛最初的4名成员,1999年该论坛开发了一种新的语言技术规范,即VoiceXML 0.9版。该论坛的发展十分迅速,2000年3月发布了VoiceXML 1.0版的技术规范。它被提交给World Wide Web联盟,该联盟的语音浏览器工作组将负责开发该技术规范的下一个版本VoiceXML 2.0。

当本书付印时,VoiceXML 2.0版的技术规范还是个草案,预计对VoiceXML 1.0版所作的修改大多与澄清有关的概念和纠正原版本的错误有关。本书将在第13章“VoiceXML的未来发展前景”中介绍技术成熟的VoiceXML 1.0版和即将发行的2.0版的变更。

1.4 VoiceXML在语音浏览器中是怎样工作的

一般来说,VoiceXML和语音浏览器通过为语音技术和内容传递提供标准界面,从而可以帮助编程人员开发因特网上运行的语音界面。尤其值得注意的是,VoiceXML提供了一个连接自动语音识别(ASR)、文本语音转换(TTS)、双音多频(DTMF)、呼叫处理以及其他技术的标准界面。

虽然VoiceXML在一定程度上降低了这些技术的复杂性,但是要编写出高质量的语音应用程序,你就必须了解它们的这些复杂性。在下面两个小节中,我们将对VoiceXML程序的输入和输出所使用的一些关键基础技术做一个概述。更加详细的内容将在第2章中介绍。

1.4.1 应用程序的输入

自动语音识别是一种使口语输入能够被程序所理解的技术。为此需要将口语中的单词分解成离散的声音小段,这种声音小段称为音素(phonemes)。然后将这些音素与应用程序开发人员或语音识别工具提供的词汇相比较。这不是可调整的或自然语言的识别方式。每给一个提示信息,VoiceXML应用程序定义一组有限的有效用户输入,其他用户的应答将不能被应用程序理解。语音识别可以通过专门的硬件或软件来实现。

双音多频(DTMF)系统通过电话网络中的硬件和软件对按键式电话的键盘输入进行翻译,实现音频播放。每个按键都能够产生一个可被识别的独一无二的音频。该系统使得用户可以使用电话键盘将数字信息输入VoiceXML应用程序,并且由程序负责接收和理解这些数字信息。

1.4.2 应用程序的输出

文本语音转换技术（TTS）能够将输出的文本转换成“语音”（实际上是一种数字音频格式）。这使得应用程序能够拥有灵活的输出方式——将开发人员没有预料到的事情告诉用户。TTS的功能与ASR基本上相反，它将文字转变为音素，这些音素具有相对应的数字声音。

另一种语音输出是用某种标准格式储存的录制的声音文件。这些声音文件格式包括WAVE和脉冲编码调制（PCM）。这些声音文件在应用程序被启用以前已经被编写和录制好了。所有录制的语音输出都必须事先确定。因此，TTS是一种更灵活的输出方式。尽管TTS技术在过去的几年里有了显著的改进，但是商用TTS语音输出的声音往往仍然比较死板，并且有时很难听懂。

这些就是VoiceXML程序员与他们的用户进行语音交互操作时使用的基本技术。

VoiceXML还为电话网络本身提供了一个非常小的界面，称为“转移（transfer）”标签。它使得程序能够将用户从当前运行的VoiceXML应用程序转到另一个电话号码。为了理解语音浏览器和转移呼叫之类的功能，就需要对有关的网络做一个说明。

1.5 高级网络结构

VoiceXML是怎样使你能够编写语音应用程序的呢？仅仅介绍一下语言技术规范显然是不够的。尽管不了解网络是如何将内容传递给用户的许多深层情况，你仍然可以编写VoiceXML应用程序，但是，如果对网络有所了解，那么你就会更好地体会你的应用程序应该遵循什么样的规则，并且有助于你开发性能更佳的VoiceXML应用程序。

VoiceXML是在XML、HTTP和TCP/IP等现行数据网络标准以及公用电话交换网（PSTN）和综合业务数字网（ISDN）中的电话标准等基础上建立起来的。

语音Web是由PSTN、因特网上的VoiceXML应用程序以及因特网与PSTN之间的VoiceXML网关等组成的。VoiceXML网关负责托管可进行语音浏览的特定硬件和软件。这些资源中的某些部分，如ASR和TTS，可以位于不同的网络组件中，并且可以进行远程访问。

在一个语音浏览器会话过程中，用户的电话呼叫通过PSTN到达一个VoiceXML网关。根据用户拨打的电话号码，该网关从因特网上下载相应的VoiceXML应用程序，并且可能将该应用程序存入高速缓存。然后该网关逐步通过VoiceXML，按照应用程序的定义与用户进行交互操作。

图1-1显示了一个典型的语音网络。

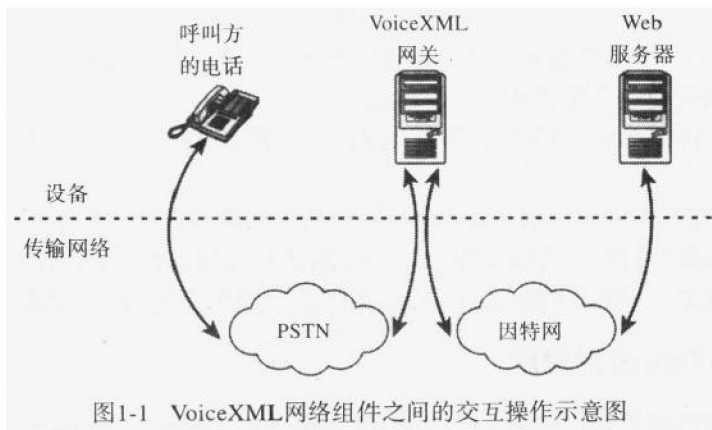
以下是对网络示意图中涉及的组件的说明：

- 呼叫方的电话——呼叫方访问VoiceXML应用程序时使用的电话。
- VoiceXML网关——在PSTN和IP之间架起桥梁的网关，并且负责托管VoiceXML浏览器、语音硬件和软件，进行被叫号码与URL的映射。
- Web服务器——这是一个在本网络中托管VoiceXML应用程序的服务器。通过编辑HTTP服务器支持的MIME（多用途因特网电子邮件扩充协议）的类型，可以从任何Web服务器来传递VoiceXML。

下面是对该网络示意图中涉及的两个网络的说明：

- PSTN——公共电话交换网，也称为简易旧式电话服务网（POPS）。这是我们大多数人在家

中使用的一种电话服务网，它能够传送我们说的话和DTMF交互操作信息，例如VoiceXML网关播放的提示信息和对呼叫方说的话作出应答。



- **Internet**——我们向因特网服务提供商（ISP）支付费用以便对其进行访问的因特网。它将网关对VoiceXML的访问请求传递给Web服务器，然后将VoiceXML返回给网关。

请注意，该网络拥有一个相当传统的结构，不符合在因特网协议上传输语音（VoIP）的运行环境。这种结构类型将在第2章中进行讲解。

关于网络的一些重点概念可以归结为，语音Web是由PSTN、因特网上的VoiceXML应用程序和因特网与PSTN之间的VoiceXML网关构成的。VoiceXML网关负责托管实现语音浏览所需要的专用硬件和软件。

1.6 VoiceXML网关

作为一名VoiceXML开发人员，你的运行期平台是VoiceXML网关。明白了VoiceXML网关的运行方法将有助于提高你的VoiceXML网页的质量，调试也更加容易。随着你获得制作VoiceXML网页的经验，你也将了解你的服务提供商所使用的VoiceXML网关的特性。

由硬件和软件构成的VoiceXML网关在PSTN与因特网之间架起了一座桥梁。这个网关包括一个VoiceXML浏览器和用于ASR、TTS和DTMF的资源，这些资源可能是硬件，也可以是软件。

下面列出了VoiceXML网关的主要功能：

- **电话**——VoiceXML首先必须接收（也称为端结）来自公共电话交换网的电话呼叫并获取与呼叫相关的信息，比如被叫号码和主叫号码。根据呼叫人拨打的电话号码，网关可以为VoiceXML应用程序提供呼叫人的位置信息，例如呼叫人所在的城市和州。如果使用电话转接标记，将用户转接到另一个电话号码，那么VoiceXML网关负责启动一个连接到转接号码的新的外接呼叫，并且将它与用户的电话呼叫连接起来。
- **URL数据库**——当网关接收到一个电话呼叫时，网关必须对被叫号码与要求的服务的URL进行核对。网关可以拥有与这个数据库相关的服务和计费系统。
- **检索VoiceXML**——当知道VoiceXML服务程序的URL的时候，网关必须从服务程序的Web主机上检索VoiceXML页和相关文件，例如录制的声音文件和语法文件。

- 翻译VoiceXML——当应用程序的VoiceXML代码和相关文件被存储在网关上之后，网关必须翻译这些代码，遍历对话框并且根据要求与ASR、TTS、DTMF以及其他服务进行交互操作。这需从Web服务器上检索其他的一些文件。
- 访问ASR和TTS——这些服务可以作为软件或者硬件被托管在VoiceXML网关上，也可以位于具备专用语音处理功能的远程服务器上。
- 高速缓存——VoiceXML网关能够将预先录制好的声音文件、语法、以及VoiceXML页存放在高速缓冲中。

IBM、朗讯、摩托罗拉和VoiceGenie 等公司直接向服务提供商和大型门户网站销售商用VoiceXML网关。其他公司生产的VoiceXML翻译软件可以与硬件和其他软件组件封装在一起，组成一个VoiceXML网关。一家名叫Speechworks的供应商则开放了它的VoiceXML翻译软件源代码。

1.7 托管VoiceXML应用程序

VoiceXML页可以像其他因特网文件那样由Web服务器来托管。任何Web服务器都可以使用，只要你能编辑MIME类型后缀映射表，在其中添加该文件类型：

Suffix: vxml Type: text/x-vxml

使用Apache、微软IIS和Sun公司的Java Web Server，你能够很容易地修改MIME类型后缀映射表，并且要托管VoiceXML文档也不成问题。但是，仅仅将你的VoiceXML应用程序放到一个Web服务器上是不够的，用户不一定能够访问它。在语音Web上，只有当你的应用程序与一个电话号码关联起来之后时，用户才能访问该应用程序。有些地方VoiceXML网关必须与你的应用程序相链接，这就必须有一个服务提供商。

VoiceXML服务提供商只具备因特网服务供应商的一半功能，即托管功能。虽然因特网服务供应商能够提供因特网接入和托管功能，但是语音Web的接入功能则由无线和本地电话公司提供。

VoiceXML服务供应商（VSP）拥有托管语音服务的专业技术和设备，包括VoiceXML网关、电话和因特网等基础设施，许多VSP托管了它们自己的语音门户并提供咨询服务。这是VSP开发的一种VoiceXML服务，用于访问VSP托管的实用应用程序或演示应用程序。

由于拨号进入VoiceXML网关的操作需要使用他人的资源，所以网关的拥有者可能要对用户扣留一定的费用，方法是向你收取访问费，或者向出现在门户上的网站收费。在目前这个施小恩谋大利的时代，很多公司将会免费托管你的VoiceXML站点，有些公司甚至向访问站点的用户提供免费接入。

负责链接你的应用程序的VSP

VSP包括BeVocal (www.bevocal.com)、HeyAnita (www.heyanita.com)、Tellme(www.tellme.com)、VoiceGenie(www.voicegenie.com)和Voxeo(www.voxeo.com)，通过它们运行的开发人员程序，你可以进行注册并且免费通过它们的VoiceXML网关来链接你的应用程序。

衡量一个VSP的优劣的方法是看他是否符合VoiceXML的技术规范。W3C的语音浏览器协会