



(12) 发明专利

(10) 授权公告号 CN 101482805 B

(45) 授权公告日 2011.06.08

(21) 申请号 200910007403.8

(22) 申请日 2004.03.05

(30) 优先权数据

10/382,121 2003.03.05 US

(62) 分案原申请数据

200410028602.4 2004.03.05

(73) 专利权人 微软公司

地址 美国华盛顿州

(72) 发明人 王冠三

(74) 专利代理机构 上海专利商标事务所有限公

司 31100

代理人 陈斌

(51) Int. Cl.

G06F 3/14 (2006.01)

G06F 17/30 (2006.01)

审查员 陈冬冰

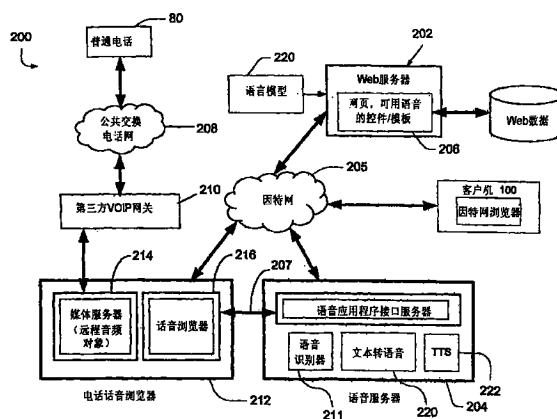
权利要求书 3 页 说明书 12 页 附图 19 页

(54) 发明名称

基于用户输入的数据呈现

(57) 摘要

提供了一种根据话音查询向用户呈现信息的方法。该方法包括从语音表述中标识第一对象和第二对象。将所述第一对象和第二对象与对应于存储信息的标记符相关联。根据所述第一对象和第二对象有选择性地呈现所述存储信息。



1. 一种在计算机系统上的基于查询向用户呈现文档中的信息的方法,其特征在于,包括:

向用户呈现文档中的数据;

从所述查询中标识第一对象和第二对象;

访问所述文档以标识与所述文档中的文本相关联的语义标记符;

将所述第一对象和与所述文档中的存储文本的第一部分对应的第一语义标记符相关联,将所述第二对象和与所述文档中的存储文本的第二部分对应的第二语义标记符相关联,其中所述第一部分和所述第二部分中的至少一个与被呈现的所述文档中的所述数据相关联;

标识存储文本的与所述第一部分和所述第二部分两者都关联的第三部分;以及

选择性地可听地呈现所述存储文本的所述第三部分。

2. 如权利要求1所述的方法,其特征在于,还包括:

接收第二查询;以及

从所述第二查询中标识包含在所述存储文本中导航的信息的导航对象。

3. 如权利要求1所述的方法,其特征在于,所述查询还包括包含用于在所述存储文本上执行选定的命令的信息的命令对象。

4. 如权利要求1所述的方法,其特征在于,所述从所述查询中标识第一对象和第二对象包括使用语言模型来标识所述第一对象和所述第二对象。

5. 如权利要求4所述的方法,其特征在于,所述从所述查询中标识第一对象和第二对象还包括使用文体控件来识别所述第一对象和所述第二对象的另选短语。

6. 如权利要求1所述的方法,其特征在于,所述存储文本是以多维结构排列的,而所述第一对象和所述第二对象中的至少一个与所述多维结构中的至少一维相对应。

7. 如权利要求6所述的方法,其特征在于,所述多维结构是一个包括多行和多列的表格,而所述第一对象包括关于特定行的信息,所述第二对象包括关于特定列的信息。

8. 如权利要求1所述的方法,其特征在于,所述选择性地可听地呈现所述存储文本的所述第三部分是基于所述第一对象和所述第二对象中的至少一个。

9. 如权利要求8所述的方法,其特征在于,所述选择性地可听地呈现所述存储文本的所述第三部分包括基于所述第一对象和所述第二对象结合存储的上下文一起呈现所述存储文本的所述第三部分。

10. 如权利要求1所述的方法,其特征在于,所述查询是语音表述。

11. 如权利要求1所述的方法,其特征在于,所述查询包括手写输入。

12. 如权利要求1所述的方法,其特征在于,所述选择性地可听地呈现所述存储文本的所述第三部分包括执行一个脚本以呈现该部分。

13. 一种在计算机系统上的基于语音查询向用户呈现文档中的信息的方法,其特征在于,包括:

向用户可听地呈现一段文本,该段包括对应于该段的各部分的各语义标记符;

在所述一段文本被可听地呈现时,从所述用户接收语音查询,并从所述查询中标识至少一个语义对象;

将所述至少一个语义对象与对应于所述段中已被可听地呈现的一部分文本的语义标

记符相关联 ; 以及

可听地呈现对应于所述语义标记符的所述段中的所述一部分文本。

14. 如权利要求 13 所述的方法, 其特征在于, 还包括分析所述段以标识所述段内的相关信息的语义标记符。

15. 如权利要求 14 所述的方法, 其特征在于, 所述一段文本是语音数据。

16. 如权利要求 13 所述的方法, 其特征在于, 所述一段文本是一个句子, 而所述各语义标记符对应于所述句子内的数据。

17. 如权利要求 13 所述的方法, 其特征在于, 所述一段文本是一行, 而所述各语义标记符对应于所述行内的各列。

18. 如权利要求 13 所述的方法, 其特征在于, 至少一个语义标记符对应于一适当的名称。

19. 如权利要求 13 所述的方法, 其特征在于, 至少一个标记符对应于一数字。

20. 如权利要求 19 所述的方法, 其特征在于, 所述至少一个标记符对应于电话号码。

21. 如权利要求 13 所述的方法, 其特征在于, 至少一个标记符对应于驾驶导向的一部分。

22. 如权利要求 13 所述的方法, 其特征在于, 所述可听地呈现所述段中的所述一部分文本包括基于来自所述查询的第一对象和第二对象结合存储的上下文一起呈现所述段中的所述一部分文本。

23. 如权利要求 13 所述的方法, 其特征在于, 所述查询是语音表述。

24. 如权利要求 13 所述的方法, 其特征在于, 所述查询包括手写输入。

25. 如权利要求 13 所述的方法, 其特征在于, 所述可听地地呈现所述段中的所述一部分文本包括执行一个脚本。

26. 一种在计算机系统向用户提供信息的方法, 其特征在于, 包括 :

从用户语音查询中标识第一对象、第二对象和命令对象 ;

访问存储信息, 并将所述第一对象和所述第二对象与对应于所述存储信息的第一部分和第二部分的标记符相关联 ;

基于与从所述用户语音查询中标识的命令对象相关联的数据操作, 按照与所述存储信息的所述第一部分和所述第二部分有关的值的函数生成数据输出, 所述数据输出不是在所述存储信息中直接找到的 ; 以及

可听地呈现所生成的数据输出。

27. 如权利要求 26 所述的方法, 其特征在于, 所述数据操作包括将所述第一部分和所述第二部分相加。

28. 如权利要求 26 所述的方法, 其特征在于, 所述数据操作包括从所述第二部分减去所述第一部分。

29. 如权利要求 26 所述的方法, 其特征在于, 所述数据操作包括基于所述第一部分和所述第二部分计算百分比。

30. 如权利要求 26 所述的方法, 其特征在于, 所述相关联包括 :

访问包含有所述存储信息的所述第一部分和所述第二部分的多个存储的信息的数据库, 其中所述多个存储的信息包括存储在所述数据库中的关系表中的值, 所述标记符对

应于所述表的列和行。

31. 如权利要求 30 所述的方法,其特征在于,所述存储信息的所述第一部分和所述第二部分包括存储在所述表中的第一数值和第二数值,所述数据操作包括用所述第一数值和所述第二数值进行的数学运算以形成结果数值。

32. 如权利要求 31 所述的方法,其特征在于,可听地呈现所生成的数据输出包括可听地呈现所述结果数值。

33. 如权利要求 30 所述的方法,其特征在于,还包括:

从所述用户话音查询标识导航对象;以及

基于所述导航对象标识与存储在所述表中的信息对应的标记符,并作为响应来可听地呈现所述存储在所述表中的所述信息。

基于用户输入的数据呈现

[0001] 本申请是申请日为 2004 年 3 月 5 日、申请号为 200410028602.4、发明名称为“基于用户输入的数据呈现”的专利申请的分案申请

技术领域

[0002] 本发明涉及在计算机系统中访问和呈现信息,尤其涉及基于用户语音输入的数据呈现。

背景技术

[0003] 许多计算机界面是计算机驱动的交互,其中用户必须遵循由计算机设定的操作流程,或学习由计算机披露的一个或多个命令。换句话说,多数计算机界面并不适应用户期望与计算机交互的方式,而是强迫用户通过特定的界面集合来交互。

[0004] 在计算机/用户界面领域进展已经允许用户通过语音命令来与计算机交互。语音门户,例如通过语音 XML(语音扩展标记语言)的运用,已经发展到能用语音输入来访问因特网内容。在这一架构中,有一个文档服务器(例如一个 Web 服务器)通过一个语音 XML 解释器处理来自客户机的请求。该 Web 服务器可以产生语音 XML 文档及回应,这些文档及回应由语音 XML 解释器处理,并以音频呈现给用户。使用经语音识别的特定语音命令,用户可以在网上巡游,并听到以音频呈现的数据。

[0005] 然而,许多向用户呈现数据的应用,例如驾驶导向、交通报告、天气报告与影片排程,并不对用户特别友好。特别是,这些应用在呈现前面已经呈现过的信息部分或存储在表格内的结构化信息部分时会有困难。举例来说,有许多服务提供驾驶导向,但是以某种用户的扩展阅读或按预先确定的步骤来提供的。其结果是,用户可能需要写下所有的驾驶导向,或继续重播完整的驾驶导向或预先确定的步骤,以便记忆相关的信息。这两种情形在许多情况下都是令人不快的。

[0006] 因此,就有以更大的灵活性来访问或呈现数据部分的需求。这样一种呈现的系统或方法将由于对用户更自然而更易于使用。

发明内容

[0007] 本发明提供一种基于语音输入来向用户呈现数据的改进的界面。在本发明的一个方面,向用户呈现信息的方法包括从语音表述(utterance of speech)来标识第一对象和第二对象。这第一和第二对象都与标记符(tag)相关联,而这些标记又与存储的信息相对应。存储的信息将根据这第一或第二对象有选择性地呈现。在一实施例中,所标识的对象可以是用于有选择性地呈现信息的查询对象、导航对象和/或命令对象。在一特定方面,存储的信息安排成一个具有多行和多列的表格。第一对象包括关于一特定行的信息,而第二对象则包括关于一特定列的信息。

[0008] 在本发明的另一方面,提供了一种方法,包括向用户呈现信息的一个片段。该片段包括了与该片断的各部分相对应的标记符。该方法进一步包括从语音表述中标识至少一个

对象,并将该对象与对应于该片断的部分的标记相关联。而后将对应于该标记的该片断的部分呈现出来。在又一实施例中,为了呈现该片断的各部分,可以用一个文本规格化器/分析器来标识该片断内的相关部分。

[0009] 其结果是,本发明提供一种适于表现多维数据与呈现在数据库中存储的信息部分的方法。提供给了用户一种基于语音输入表现数据的更为自然的界面。举例来说,用户可以查询一个表格中的单个单元,或基于存储的数据创建一个双向对话。

附图说明

- [0010] 图 1 是一数据表现系统的框图;
- [0011] 图 2 是一计算装置操作环境的平面图;
- [0012] 图 3 是一图 2 所示计算装置的框图;
- [0013] 图 4 是一种电话机的平面图;
- [0014] 图 5 是一种通用计算机的框图;
- [0015] 图 6 是一种用于客户机/服务器系统架构的框图;
- [0016] 图 7 是一种语音识别与理解模块的框图;
- [0017] 图 8 是一种数据呈现模块的框图;
- [0018] 图 9 所示是一股票价格表和用于呈现该表格内数据的对象;
- [0019] 图 10A-10C 包含用于呈现图 9 中表格的示例性代码;
- [0020] 图 11 所示是驾驶导向表和用于呈现该表格内数据的对象;
- [0021] 图 12 所示是销售数据表和用于呈现该表格内数据的对象;
- [0022] 图 13 所示是一个文本段落和用于呈现该文本段内数据的对象;
- [0023] 图 14A-14D 包含用于呈现图 13 中文本段落的示例性代码;
- [0024] 图 15 是一条语音邮件报文和用于呈现该语音邮件报文内数据的对象。

具体实施方式

[0025] 图 1 所示是基于语音输入而呈现数据的数据表现系统 10 的框图。系统 10 包括接口模块 12、言语识别与理解模块 14 和数据呈现模块 16。用户以话音查询形式向语音接口模块 12 提供输入。语音接口模块 12 收集来自用户的语音信息并为其提供指示信号。在输入语音由语音接口模块 12 收集后,语音识别与理解模块 14 用语音识别器识别该语音并标识对象,诸如关于用户希望系统 10 呈现信息的关键词或关键短语。这些对象由数据呈现模块 16 用来从数据库 18 中获取数据。一旦相关信息被用这些对象从数据库 18 中标识出来后,相关信息就可以被呈现给用户。数据呈现模块 16 的输出可以是不同的形式,包括音频和/或视频输出。

[0026] 倘若基于话音查询来广泛地讨论呈现数据,一般性地描述可用于上述系统 10 的计算装置可能是有益的。如本领域的技术人员所了解,系统 10 的部件可以位于单台计算机内,也可分布在某个使用网络连接及协议的分布式计算环境中。

[0027] 现在参照图 2, 30 表示一种移动装置的示例性形式,如数据管理装置(PIM、PDA 等)。然而,可预期本发明也可以用以下讨论的其它计算装置来实现。例如,电话和/或数据管理装置也将从本发明中获益。这类装置与现有的便携个人信息管理装置和其它便携电

子装置相比,将会具有增强的效用。

[0028] 图 2 给出了一数据管理移动装置 30 的示例性形式。该移动装置 30 包括一个外壳 32,并具有一个用户接口,包括一个显示屏 34,使用一个与触笔 33 联接的触摸显示屏。触笔 33 用来在指定坐标按压和触碰显示屏 34,以选择某个字段,或有选择性地移动光标的起始位置,或通过如姿势或手写来提供命令信息。可替换地或额外地,在装置 30 上可以包括一个或多个按钮 35,用来导航。而且,还可以提供其它输入机制,如滚轮、滚筒等。然而应当注意,本发明不应被这些输入机制的形式所限制。举例来说,输入的其它形式还可以包括某种视觉输入,如通过计算机视觉。

[0029] 现在参照图 3,一幅框图例示了组成移动装置 30 的功能部件。中央处理单元 (CPU) 50 实现软件控制功能。CPU 50 与显示屏 34 连接,这样根据控制软件所产生的文本和图标就可以显示在显示屏 34 上。可以将扬声器 43 与 CPU 50 相连,一般还带有数模转换器 59,以提供音频输出。由用户下载和输入到移动装置 30 中的数据存储在 CPU 50 双向相连的某个非易失性可读 / 写随机访问存储器 54 中。随机访问存储器 (RAM) 54 提供对将由 CPU 50 所执行指令的易失存储,以及对临时数据如寄存器值的存储。配置选项或其它变量的缺省值存储在只读存储器 (ROM) 58 中。ROM 58 还可以用于为装置存储控制移动装置 30 基本功能和其它操作系统核心功能 (如将软件模块载入 RAM 54) 的操作系统软件。

[0030] RAM 54 还可以以与 PC 上用来存储应用程序相类似的硬盘驱动器方式,用来作为代码的存储器。应当注意,尽管非易失性存储器是用来存储代码,但可替换地也可以在未用于代码执行的易失性存储器中存储。

[0031] 该移动装置可以通过与 CPU 50 相连的无线收发器 52 来发送 / 接收无线信号。如果需要,也可以提供可选的通讯接口 60,以从计算机 (如桌面计算机) 或联网网络直接下载数据。因此,接口 60 可以包括多种形式的通讯装置,如红外链路、调制解调器、网卡等。

[0032] 移动装置 30 包括麦克风 29、模数 (A/D) 转换器 37、以及存储在存储器 45 中的可选的 (语音、DTMF、手写、姿势或计算机视觉) 识别程序。举例来说,回应于来自装置 30 用户的音频信息、指令或命令,麦克风 29 提供由 A/D 转换器 37 数字化了的语音信号。语音识别程序可以在这些数字化了的语音信号上执行规格化和 / 或特性析取函数,以获取中间语音识别结果。使用无线收发器 52 或通讯接口 60,语音数据可以被发送给下面所讨论并在图 6 的架构中所示的某个远端识别服务器 204。而后识别结果被返回给移动装置 30,以便在其上 (如以视频或音频) 呈现,并当在 Web 服务器与移动装置 30 以客户机 / 服务器关系工作时,最终发送给 Web 服务器 202 (图 6)。对于其它形式的输入也可以运用类似的处理。例如,手写输入可以在装置 30 上带或不带预处理地被数字化。象语音数据一样,这种形式的输入也可以被发送给识别服务器 204 以便识别,而识别结果被返回给装置 30 和 / 或 Web 服务器 202 至少其中之一。同样地,DTMF 数据、姿势数据和视频数据也可以被类似地处理。依据输入的形式,装置 30 (以及下面所讨论的其它形式的客户机) 可以包括必要的硬件,如用于视频输入的摄像机。

[0033] 图 4 是一幅移动电话 80 的示例性实施例的平面图。该电话 80 包括显示屏 82 和键盘 84。一般而言,图 3 的框图可应用于图 4 的电话,尽管可能需要执行其它功能所必需的额外电路。例如,对于图 3 的实施例将需要一个作为电话运作所必需的收发器;然而,这样的电路不涉及本发明。

[0034] 除上述的便携或移动计算装置外,还应当理解本发明也可以用于多种其它计算装置,如通用桌面计算机。例如,当其它常规输入装置,如全字母-数字键盘太难以操作时,本发明可以使有身体障碍的用户将文本输入计算机或其它计算装置。

[0035] 本发明也可运作于多种其它通用或专用计算系统、环境或配置。可以适用于本发明的众知计算系统、环境和 / 或配置的例子包括,但不限于,(不带任何显示屏的)寻常电话、个人计算机、服务器计算机、手持或膝上装置、平板计算机、多处理器系统、基于微处理器的系统、机顶盒、可编程消费电子产品、网络 PC、小型计算机、大型计算机、包括以上系统或装置之一的分布式计算环境等。

[0036] 以下是对在图 5 中所示的通用计算机 120 的简要描述。然而,该计算机 120 仍然只是适用的计算环境的一个示例,而并非提出对本发明的使用或功能的任何限制。计算机 120 也不能被解释成对其中所图示的部件之一或其组合有任何依赖或要求。

[0037] 本发明可以以计算机可执行指令的一般上下文来描述,如由计算机执行的程序模块。一般而言,程序模块包括执行特定任务或实现特定抽象数据类型的例程、程序、对象、组件、数据结构等。本发明还可以在分布式计算环境中实现,在该环境中任务是由通过某种通讯网络相联接的远端处理装置来执行。在分布式计算环境中,程序模块可以位于包括存储器装置的本地或远端计算机存储媒介中。由这些程序或模块所执行的任务将在下面借助于附图来描述。本领域技术人员可以将这些描述和附图实现成处理器可执行指令,而这些指令可以写以任何形式的计算机可读取媒介上。

[0038] 参照图 5,计算机 120 的部件可以包括,但不限于,处理单元 140、系统存储器 150、以及将多个系统部件包括系统存储器连接到处理单元 140 的系统总线 141。系统总线 141 可以是包括存储器总线或存储器控制器、外围装置总线、使用多种总线架构之一的局部总线的几种总线架构之一。举例而言,但非限制,这样的总线架构包括工业标准架构 (ISA) 总线、通用串行总线 (USB)、微通道架构 (MCA) 总线、增强 ISA (EISA) 总线、视频电子标准协会 (VESA) 局部总线、以及也称为 Mezzanine 总线的外围部件互连 (PCI) 总线。计算机 120 一般包括多种计算机可读取媒介。计算机可读取媒介可以是能由计算机 120 所访问的任何可用的媒介,包括易失和不易失媒介、可移动和不可移动媒介。举例来说,但非限制,计算机可读取媒介可以由计算机存储器媒介和通讯媒介组成。计算机存储器媒介包括以任何用以存储信息,如以计算机可读取指令、数据结构、程序模块或其它数据的方法或技术实现的易失和不易失媒介、可移动和不可移动媒介。计算机存储器媒介包括,但不限于, RAM、ROM、EEPROM、闪存或其它存储器技术、CD-ROM、数字多功能光盘 (DVD) 或其它光盘存储器、盒式磁带、磁带、磁盘存储器或其它磁存储装置、或可用以存储所需要信息并能由计算机 120 访问的任何其它媒介。

[0039] 通讯媒介一般将计算机可读取指令、数据结构、程序模块或其它数据实现在某种调制的数据信号中,如某种载波或其它传输机制,并包括任何信息传递媒介。术语“调制的数据信号”意指某种信号,其一个或多个特征可以某种方式设置或改变,以便将信息编码到该信号中。举例而言,但非限制,通讯媒介包括有线媒介如有线网络或直线连接,以及无线媒介如声音、FR、红外或其它无线媒介。以上任意种类的组合也应当包括在计算机可读取媒介的范围内。

[0040] 系统存储器 150 包括易失和 / 或非易失存储器形式的计算机存储器媒介,如只读

存储器 (ROM) 151 和随机访问存储器 (RAM) 152。在 ROM 151 中一般存储了基本输入 / 输出系统 153 (BIOS), 含有如在启动时帮助在计算机 120 内各元素间传递信息的基本例程。RAM 152 一般包含可由处理单元 140 立即访问和 / 或当前在处理单元 140 上执行的数据和 / 或程序模块。举例而言, 但非限制, 图 5 例示了操作系统 154、应用程序 155、其它程序模块 156、以及程序数据 157。

[0041] 计算机 120 还可以包括其它可移动 / 不可移动的易失 / 不易失计算机存储媒介。仅举例而言, 图 5 例示了读写不可移动不易失磁媒介的硬盘驱动器 161、读写可移动不易失磁盘 172 的磁盘驱动器 171、以及读写可移动非易失光盘 176 如 CDROM 或其它光媒介的光盘驱动器 175。其它可以用于该示例性操作环境的可移动 / 不可移动、易失 / 不易失计算机存储媒介包括, 但不限于, 盒式磁带、闪存卡、数字万用盘、数字录影带、固态 RAM、固态 ROM 等。硬盘驱动器 161 一般通过不可移动存储器接口如接口 160 来连接到系统总线 141, 而磁盘驱动器 171 和光盘驱动器 175 一般由可移动存储器接口如接口 170 来连接到系统总线 141。

[0042] 上面讨论并在图 5 中例示的驱动器及与之相关联的计算机存储媒介为计算机 120 提供计算机可读取指令、数据结构、程序模块和其它数据的存储。例如, 在图 5 中, 硬盘驱动器 161 被例示成存储着操作系统 164、应用程序 165、其它程序模块 166、以及程序数据 167。请注意这些部件可以相同也可以不同于操作系统 154、应用程序 155、其它程序模块 156、以及程序数据 157。这里为操作系统 164、应用程序 165、其它程序模块 166、以及程序数据 167 赋予不同的号码, 是为了例示至少它们是不同的拷贝。

[0043] 用户可以通过输入装置如键盘 182、麦克风 183、以及定位装置 181 (如鼠标、轨迹球或触摸板), 将命令和信息输入计算机 120。其它输入装置 (未示出) 可以包括游戏杆、游戏手柄、卫星碟天线、扫描仪等。这些或其它输入装置通常通过一个与系统总线相连的用户输入接口 180 来连接到处理单元 140, 但也可以由其它接口和总线架构来连接, 如并口、游戏口或通用串行总线 (USB)。监视器 184 或其它类型的显示装置也通过接口如视频接口 185 来连接到系统总线 141。除监视器外, 计算机还可以包括其它外围输出装置如扬声器 187 和打印机 186, 这些可以通过输出外围接口 188 来连接。

[0044] 计算机 120 可以使用与一台或多台远端计算机 (如远端计算机 194) 的逻辑连接, 在联网环境中运行。远端计算机 194 可以是个人计算机、手持装置、服务器、路由器、网络 PC、对等装置或其它公共网络节点, 而且一般包括在上面关于计算机 120 所描述的多个或所有元件。在图 5 中所描述的逻辑连接包括局域网 (LAN) 191 和广域网 (WAN) 193, 但也可以包括其它网络。在办公室范围、企业范围计算机网络、内联网和因特网中, 这样的联网环境是很普通的。

[0045] 当在 LAN 联网环境中使用时, 计算机 120 通过网络接口或适配器 190 连接到局域网 191。当在 WAN 联网环境中使用时, 计算机 120 一般包括调制解调器 192 或其它用于在广域网 193 如因特网上建立通信的装置。调制解调器 192 可以是内置的或外置的, 可以通过用户输入接口 180 或其它合适的机制连接到系统总线 141。在联网环境中, 关于计算机 120 或其各部分所描述的程序模块可以存储在远端存储装置中。举例而言, 并非限制, 图 5 将远端应用程序 195 例示为驻留在远端计算机 194 上。可以了解, 所示出的网络连接是示例性的, 还可以使用在计算机间建立通讯链路的其它手段。

[0046] 图 6 例示了用于基于 Web 的识别与数据呈现的架构 200, 这也是本发明的一个示

例性环境。一般,存储在 Web 服务器 202 中的信息可以通过某个客户机 100,如移动装置 30 或计算机 120(在此代表基于输入形式而具有显示屏、麦克风、摄像机、触摸屏等的其它形式计算装置)来访问,或通过有声地请求信息的电话 80 来访问,或通过响应按键由电话 80 产生音调访问,而来自 Web 服务器 202 的信息则仅有声地提供回用户。

[0047] 在这一实施例中,架构 200 被统一,其中:无论信息是通过客户机 100 还是通过电话 80 使用语音识别而获取的,只要一台识别服务器 204 就可以支持任一模式的操作。而且,架构 200 使用众所周知的标记语言的扩展(如 HTML、XHTML、cHTML、XML、WML 等)来运作。因此,存储在 Web 服务器上的信息也就可以用这些标记语言中可找到的众所周知的 GUI 方法来访问。通过使用众所周知的标记语言的扩展,在 Web 服务器 202 上的编著就更为容易,而现存的传统应用也可以更容易地修改成包括话音识别。

[0048] 一般而言,客户机 100 执行由 Web 服务器使用浏览器提供的 HTML 页、脚本等(在 206 示出)。当需要话音识别时,举例来说,语音数据(可以是数字化了的音频信号或其中这些音频信号则已先由客户机 100 按上面所讨论的那样预处理过的语音特征)将和一个在语音识别时使用的语法或语言模型 220 的指示一起被提供给识别服务器 204,而该模型可以由客户机 100 来提供。可替换的是,语音服务器 204 也可以包括该语言模型 220。识别服务器 204 的实现可以利用多种形式(其中一种已图示),但一般包括识别器 211。如果需要或合适,识别的结果将提供回客户机 100 以作本地呈现。如果需要,也可以用“文本转语音”模块 222 来向客户机 100 提供说出来的文本(spoken text)。在通过识别及所使用的任何图形用户界面编辑信息后,客户机 100 将该信息发送给 Web 服务器 202 以作进一步处理,并且如果必要的话,接受进一步的 HTML 页/脚本。

[0049] 如图 6 中所示,客户机 100、Web 服务器 202 和识别服务器 204 是通过网络 205 共同地联接在一起,但可分别寻址,网络 205 在这里是广域网,如因特网。因此,并没有必要要求这些装置都在物理上彼此相邻。特别是,Web 服务器 202 不是必须要包括识别服务器 204。在这种方式下,Web 服务器上的创作可以关注于所要做的应用本身,作者不需要了解识别服务器 204 的错综复杂。相反,识别服务器 204 可以独立地设计并联接到网络上,因而可以更新与改进而不需要在 Web 服务器 202 上作进一步的改变。Web 服务器 202 还可以包括某种编著机制,动态地产生客户端的标记和脚本。在另一实施例中,Web 服务器 203、识别服务器 204 和客户机 100 可以根据所实现的机器的性能而组合。例如,如果客户机 100 包括一台通用计算机如一台个人计算机,则该客户机可以包括识别服务器 204。同样,如果需要,Web 服务器 202 和识别服务器 204 也可以组合到一台机器中。

[0050] 通过电话 80 对 Web 服务器 202 的访问包括将电话 80 连接到某个有线或无线网络 208,而该网络又将电话 80 连接到第三方网关 210。网关 210 将电话 80 联接到电话语音浏览器 212。电话语音浏览器 212 包括提供电话接口的媒体服务器 214 和话音浏览器 216。象客户机 100 一样,电话语音浏览器 212 从 Web 服务器 202 接收 HTML 页/脚本。在一实施例中,HTML 页/脚本的形式与提供给客户机 100 的 HTML 页/脚本相似。在这种方式下,Web 服务器 202 就不需要独立地支持客户机 100 和电话 80,甚至独立地支持标准的 GUI 客户机。相反,可以使用共同的标记语言。而且,象客户机 100 一样,来自电话 80 发送的有声信号的话音识别也是从话音浏览器 216 提供给识别服务器 204,或者是通过网络 205,或者是通过专线 207,例如使用 TCP/IP。Web 服务器 202、识别服务器 204 和电话语音浏览

器 212 可以在任何合适的计算环境中实现,如在图 5 中所示的通用桌面计算机。

[0051] 描述了在系统 10 中发挥作用的各种环境和架构之后,将提供对系统 10 多种部件和功能的更详细的描述。图 7 例示了一幅语音识别与理解模块 14 的框图。从语音接口模块 12 所接收到的输入语音被发送到语音识别与理解模块 14。语音识别与理解模块 14 包括识别引擎 306,该引擎有相关联的语言模型 310。识别引擎 306 使用语言模型 310 来标识可能的表面语义结构,以表示各自的输入。识别引擎 306 根据输入语音提供至少表面语义输出对象。在某些实施例中,识别引擎 306 能够为每个可选择的结构提供一个以上的可选择表面语义对象。

[0052] 尽管在图 7 中提供的是语音输入,但本发明还可以用于手写识别、姿势识别或图形用户界面(其中用户以键盘或其它输入装置来交互)。在这些实施例中,语音识别器 306 将由在相关领域中所熟知的合适的识别引擎所替代。对于图形用户界面而言,会有某个语法(具有语言模型)与用户输入相关联,例如通过某个输入框。因此,用户的输入就能以一致的方式来处理,而不必根据输入的方式而作大的改动。

[0053] 对于基于语言的用户输入,如语音和手写,由识别引擎 306 所使用的语言模型 310 可以是任何一种已知的随机模型。举例来说,该语言模型可以是某种 N 阶(N-gram)模型,这种模型根据输入中前面的一组 N 个词来模型化在某种语言中一个词的概率。该语言模型还可以是某种上下文无关语法,该语法将语义和 / 或句法信息与特定的词和短语相关联。在本发明的另一实施例中,使用了一种将 N 阶(N-gram)语言模型和上下文无关语法组合起来的统一的语言模型。在该统一的语言模型中,语义和 / 或句法标记符被处理成词的位置值,并对于每个假设的词与记号的组合来计算 N 阶概率。

[0054] 语言模型 310 能够根据为数据呈现模块 16 将相关信息作为所提供对象的一个函数来呈现所必要的信息而产生一个分等级的表面语义结构。在一实施例中,将分析输入语音来标识在输入文本中的各种语义记号或对象。这些对象是从在语言模型 310 中所找到的一组对象中标识出来。一般而言,这些对象表示了由数据呈现模块 16 呈现信息所使用的信息。如下面所描述的那样,这些对象可以包括查询对象、导航对象和 / 或命令对象。查询对象包含了关于数据库 18 中所存储信息的信息。导航对象包含了用来传过存储信息导航的信息,而命令对象可以根据存储信息执行各种命令。

[0055] 语音识别与理解模块 14 还可以使用一个文体控件 312 来对输入语音中所标识的对象识别另选的短语。文体控件 312 与语言模型 310 相关联,帮助向数据呈现模块 16 提供相关的对象。在图 6 所示的环境中,关于文体控件 312 的信息可以由某个使用编著工具(如华盛顿 Redmond 的 Microsoft 公司的 ASP.NET)的应用编著者在 Web 服务器 202 上来实现。可替换的是,也可以使用其它编著工具如 JSP、J2EE、J2SE 或 J2ME 等。举例来说,短语“到下一转弯的距离是多少?”可以被“文体化”成象“到下一个转弯有多远?”这样的短语。而且,“下一转弯的方向是什么?”可以被重新短语化成“下一转弯是哪个方向?”或“我下一个转弯是朝哪个方向?”。如此,文体控件 312 就可以用来标识数据库 18 内的相关数据,并标识提供给用户的合适答案。

[0056] 当用户提供了语言模型不认识的语音时,系统可以提示用户重复输入。但是,如果系统根据输入中的语义信息没有或缺乏相关的信息,则系统可以执行某个合适的帮助例程,告诉用户可用的选项。

[0057] 图 8 例示了一幅数据呈现模块 16 的详细框图。数据呈现模块 16 包括命令操作器模块 602、文本分析器 / 规格化器模块 604、数据库接口模块 606、答案生成器模块 607、音频接口模块 608、视频接口模块 610 和“文本转语音”模块 612。数据呈现模块 16 接收来自语音识别与理解模块 14 的对象,并将相关信息的某个输出(音频和 / 或视频的)提供给用户。如早先所提到的那样,关键词或短语由语音识别与理解模块 14 来标识,并提供一个对象作为其函数。数据呈现模块 16 解读从语音识别与理解模块 14 接收到的对象,以便使用数据库接口 606 从数据库 18 中检索或析取数据。数据库接口 606 包括关于数据库 18 中存储数据的结构或模式的信息。应注意,数据库接口 606 可以是一种能够从各种不同来源(如从本地计算机或从位于广域网中的某台 Web 服务器)访问数据的通用模块。为析取相关信息,数据呈现模块 16 将从语音识别与理解模块 14 接收到的对象和与数据库 18 中存储信息相应的标记符或标识符关联起来。

[0058] 在某些情况下,存储在数据库 18 中的数据已经包括了各种与数据库 18 中的信息类型或信息结构相应的标记符或标识符。在其它情况下,可以用文本分析器 / 规格化器 604 来产生标记符或用其它方法标识数据中的相关信息。在将相关信息呈现给用户之前,可以对数据进行额外的处理。例如,可以用命令操作器 602 来根据接收到的对象处理从数据库 18 获得的数据的各种组合。

[0059] 一旦根据用户的请求处理了相关信息后,数据就被送到答案生成器 607。答案生成器 607 可以对用户提供的输入产生某个合适的答案。尔后答案生成器 607 将数据送到音频接口 608 和 / 或视频接口 610,以呈现给用户。在音频接口 608 内的某个“文本转语音”模块 612 可以用来将数据有声地呈现。

[0060] 图 9 示意性地例示了数据库 18 中的一个可以通过查询有选择性地呈现给用户的表格 650。表格 650 显示了在 2002 年中每个季度末各个公司的股票收盘价格。表格 650 包括了存储在行 652 的公司名称、对应 2002 年的每个季度的列 654,以及各列与行中的股票价格 656。与表格 650 相关联的标记符对应于这些列与行。查询对象 660 和导航对象 662 由语言模型 310 定义,用以呈现表格 650 中的数据。

[0061] 为了有选择性地呈现表格 650 中的数据,用户提供一个查询,包括查询对象 660 和 / 或导航对象 662。该查询由识别与理解模块 14 解读,以标识相关的查询和导航对象。尔后这些对象被和与这些列与行相对应的标记符相关联。

[0062] 查询对象 660 可以用来呈现来自表格 650 中某个特定单元的信息。举例来说,语音查询可能是:“Microsoft 在第二季度的股票收盘价格是什么?”。在这一情况下,语音识别与理解模块 14 会将查询对象“Microsoft”和“2 季度”提供给数据呈现模块 16。利用这些对象,数据呈现模块 16 就将这些对象与数据库 18 的标记符相关联,以确定所要呈现的合适单元(显示成表格 650 中的阴影单元)。在这一情况下,“Microsoft”是一个包含关于对象<公司名称>的信息的查询对象,而“2 季度”是一个包含关于对象<季度>的信息的查询对象。

[0063] “Microsoft”与对应于表示为“MSFT”的行的标记符相关联,而“2 季度”与对应于表示为“Q2”的列的标记符相关联。当这些对象与合适的标记符相关联后,股票价格“54.12”就被提供给答案生成器 607。一个答案就利用这些相关信息而产生了,并被使用音频接口 608 和 / 或视频接口 610 呈现给用户。举例来说,所呈现的答案可能是“Microsoft

在第二季度的股票收盘价格是五十四美元十二美分”。

[0064] 在本例中,答案生成器 607 接收数值“54.12”,并将该数值与存储的上下文组合使用,以呈现所检索到的数据。在本例中,所存储的上下文是“<公司名称>在<季度>的股票收盘价格是<结果>”,其中<季度>和<结果>也已被规格化。用来呈现所检索到数据的上下文可以视必要而单独与该数据的标记符或标识符相关联,和/或可作为对象的某个函数。如果要提供视频输出,答案生成器 607 可以提供如何可视地显示所检索到数据的指示。

[0065] 此外,查询对象 660 可以包括能呈现一个整行或一个整列的对象。举例来说,用户可能问:“Microsoft 在 2002 年所有季度的股票收盘价格是什么?”。在这一情况下,数据呈现模块 16 将向用户呈现 2002 年中 Microsoft 股票价格的每个数值。

[0066] 导航对象 662 可以用来让用户在表格 650 中相对于表格中的某一位置而导航。举例来说,用户在查询了 Microsoft 在 2 季度的股票收盘价格后可能问:“Microsoft 在下一季度的股票收盘价格是什么?”。在这一情况下,语音识别与理解模块 14 将标识对象“Microsoft”和“下一季度”。这些对象将与对应“Microsoft”行的标记符以及下一列的标记符如列“Q3”相关联。其结果是,在 Microsoft 行中对应于下一季度的数据将被呈现。

[0067] 可以用各种语音应用程序接口来实现本发明。例如,有一种这样的接口就是如华盛顿 Redmond 的 Microsoft 公司所开发的 SAPI。而且,本发明用标记语言扩展如语音应用语言标记 (SALT) 来实现。SALT 是一个正在发展的标准,用于实现从例如个人计算机、电话、平板 PC 及无线移动装置对信息、应用和 Web 服务的访问。SALT 扩展了现有的标记语言如 HTML、XHTML 和 XML。SALT 1.0 规范可以在线于 <http://www.SALTforum.org> 找到。应当注意,SALT 可以根据用户的输入,例如从语音服务器的输入提供语义信息,而这样的信息就形成了提供给数据呈现模块 16 的对象。如下面所进一步描述的那样,对 SALT 扩展或类似扩展的使用能提供对事件驱动的用户交互的支持,以有选择性地呈现数据。

[0068] 图 10A-10C 提供了使用 SALT 如上面所述那样呈现表格 650 中数据的示例性 XML 代码。如在图 10A 中所例示的那样,该代码包括一个头部部分 670、一个数据部分 672 和一个输入部分 674。头部部分 670 包括多种用于初始化和建立网页或应用的元素的信息。数据部分 672 以多种标记符表示了表格 650 的数据。举例来说,数据部分 672 包括了对应<公司>的标记符 676(指示一行)、对应<名称>的标记符 677 和对应<Q2>的标记符 678,<名称>和<Q1><Q2>等代表列。尽管其中所示的数据部分 672 包括了所要呈现的数据,但数据部分 672 也可以通过使用如统一资源定位器 (URL) 来包括指向具有这些信息的其它位置的链接。输入部分 674 定义了预期会来自用户的多种输入。

[0069] 图 10B 接续了用于呈现表格 650 中数据的代码。在图 10B 中,用标记符“SALT”表示了多种语音应用标记符。举例来说,标记符包括“听”标记符 680、“语法”标记符 682 和“提示”标记符 684 与 686。听标记符 680 用于语音输入。该听标记符配置了一个语音识别器,执行识别并处理语音输入事件。语法标记符 682 用来指定在识别中所使用的语法。以这种方式,语法 682 标识一种语言模型。在本例中,语法的规则部分 688 对表格 650 中的各种公司名称定义,而规则部分 690 则对表格 650 中的每个季度定义。提示标记符 684 和 686 用来指定系统输出,即上面所述的上下文。这些提示标记符可以是简单文本、语音输出标记、变量值、到音频文件的链接,或它们的组合。如下面所描述的那样,还可以用函数和/

或脚本方法来格式化所检索的数据。提示 684 根据用户的请求产生一个答案,并充当图 8 中所示的答案生成器 607。提示 686 要求用户输入一个查询。

[0070] 图 10C 从图 10B 继续代码,包括一个根据用户的话音查询呈现相关信息的脚本 692。该脚本 692 标识了所要呈现的相关单元,并调用提示 684,根据所标识的对象以及这些对象与对应数据部分 672 中数据的标记符间的关联来进行呈现。本例还例示了事件支持和嵌入的脚本页面存储,在激活了对象的识别与标识后,就在脚本部分 692 中调用或执行某个函数,以有选择性地呈现数据。

[0071] 数据呈现模块 16 在计算机与用户之间创建对话时也特别有用。在用户希望根据请求来检索存储在某个数据库中的信息部分的情况下,一个对话将特别有用。有一个这样的情况就是呈现驾驶导向。图 11 例示了一个包括简单驾驶导向的表格 700。表格 700 安排成多个行 702 和多个列 704。每一行 702 代表驾驶导向中的一个转向,而每一列 704 代表关于每个转向的特定信息。在 706 所示的额外信息也可以与表格 700 相关联。这些额外信息 706 被显示为旅程的总量,但也可以包括其它信息或对其它信息的链接。在某一实施例中,就提供了关于附近商业如银行和饭店的信息。还有多个查询对象 708 和多个导航对象 710 也与表格 700 相关联。

[0072] 在向用户呈现驾驶导向时,数据呈现模块 16 可能缺省呈现第一行(转向)的信息。数据呈现模块 16 可以编程为向用户呈现第一个转向的所有或部分信息。例如,给了第一行方向信息后,答案生成器 607 可以向用户有声地呈现:“在 0.5 英里处向左转向 Concord 大道”。用户尔后可能询问关于该转向的进一步信息,如“我应该找什么路标?”。可替换地,用户也可能要求重复转向的部分。例如,用户可能问:“我要转向哪个方向?”。在这一情况下,方向对象将与当前方向的标记符相关联,即为“向左”。数据呈现模块 16 从表格 700 检索该相关信息,并呈现一个合适的答案,例如“转左”,这里“左”是从第一行第一列取得的。当用户希望听到下一个转向时,用户可以提供一个查询,例如“下一个转向是什么?”。

[0073] 利用导航对象 710,数据呈现模块 16 可以呈现对于当前位置的转向相关信息。举例来说,用户可能问:“下一个转向的街名是什么?”。导航对象“下一个”将与在表格 700 中给定当前位置的下一个转向(即行)的标记符相关联,查询对象街名将与适当的列相关联,而相关信息将被呈现。

[0074] 在任何时候,用户可以用某个提供相应对象的适当查询来访问表格 700 的任何部分。而且,用户可以通过某个与对象<总量>相关联的适当查询来访问总距离和大致旅行时间 706。可替换的是,一个查询可以根据当前位置和来自用户的输入来请求一组新的驾驶导向。举例来说,用户可能说:“请将我带到最近的墨西哥饭店”。这一输入将被解读成根据当前位置和提供最近的墨西哥饭店的数据来产生一组新的驾驶导向。因此,可以根据这一信息扩展与驾驶导向相关联的语言模型,来识别多种查询、导航或命令对象,并且如果必要,就执行能获取包含在远端数据库(用于访问远端数据)中的新数据的代码(如脚本)。该系统还可以获取新的语言模型,来有选择性地呈现新数据。在某一实施例中,前面从如某一标记页或其它代码所呈现的数据可以与记录下的当前位置一起存储起来,这样在呈现完新信息后,系统就可以回去从当前位置呈现前面的信息(即表格 700)。

[0075] 数据呈现模块 16 还可以用来执行特定的命令。图 12 示意性地将数据例示为表格

750 和 752, 分别包括 2001 年和 2002 年中的产品销售数据。除了象上面所述的那样查询单独的单元以及在表格中巡航以外, 用户还可以利用处理表格 750 和 752 中数据的命令来请求信息以便呈现所描述信息。在从表格 750 和 752 呈现数据时, 查询对象 760、导航对象 762 和命令对象 764 都要使用。利用命令对象 764, 用户可以根据表格 750 和 752 中的信息有选择性地呈现相关信息, 并根据这些信息执行命令。

[0076] 举例来说, 在使用<比较>对象时, 用户可能请求: “请给我部件 1001 在 2001 年第一季度和 2002 年第一季度地销售数据”。对于这个查询, 数据呈现模块 16 将带或不带上下文地有选择性地呈现数值“3048.26 美元”和“4125.06 美元”。在某一实施例中, 这些数值以并排关系显示以便比较, 并有声地呈现出来。

[0077] 使用命令对象 764, 命令操作器 602 还能根据用户的请求来计算数据。举例来说, 用户可能问: “请将部件 1001 在 2002 年第一季度和 2002 年第二季度的销售相加”。这一命令使用也是从用户输入中标识出来的<相加>对象。在这一情况中, 数据库接口 606 将析取部件 1001 在 2002 年相关季度的信息数值, 并将这些相关数据送给命令操作器 602。而后命令操作器 602 将每个数值相加在一起, 并将结果送给答案生成器 607, 而答案生成器 607 用音频接口 608 和 / 或视频接口 610 来呈现这些数据。命令操作器 602 还可以相加多于两个的数值, 例如一整行的信息。基于特定的应用, 还可以使用其它命令。例如, <相减>和<百分比>可以基于两个或多个数值来呈现数值。

[0078] 数据呈现模块 16 还可以有选择性地呈现未结构化的数据, 例如一段文本, 而这段文本在数据库 18 中可能是来自某个有声文件, 或经合适转换的手写输入。图 13 例示了有关某股票市场概要的一段文本 800。查询对象 802 和导航对象 804 被定义来有选择性地根据用户话音输入呈现段落 800。为了有选择性地呈现段落 800, 必须将多种标记符与段落 800 内的相关信息相对应。在某一实施例中, 用文本规格化器 / 分析器 604 来标识段落 800 中的相关部分, 并根据这些相关部分产生多种标记符。举例来说, 规格化器 / 分析器 604 可以标识句子 (类似于上面所解释的表格中的行)、数字、公司名称等。而处理可以包括为数据各部分确定语义信息。

[0079] 在段落 800 被预处理以标识相关标记符后, 就可以呈现段落 800 了。起初, 数据呈现模块 16 开始呈现文本的第一个句子。在用户的静默或识别出某个“下一个”导航对象后, 数据呈现模块将开始呈现下一个句子。

[0080] 用户还可以请求呈现段落 800 的某个部分。举例来说, 用户可能使用某个查询如“最近的股票指数是什么?”来请求重复最近的股票指数。在语音识别与理解模块 14 标识出<股票指数>对象时, 数据呈现模块 16 将该对象与段落 800 中的某个标记符相关联。例如, 在呈现了段落 800 的第一个句子后, 数据呈现模块 16 将该股票指数对象与某个对应于“标准普尔五百股票指数”的标记符相关联。这样, 在呈现一段信息 (即一个句子) 后, 可以根据用户的话音输入而呈现该段的某个部分。应当理解, 该段落的任何部分, 都可以利用某个提供与访问所需信息相对应的对象的合适查询而获得。这种处理未结构化数据而后让用户提供查询、导航和命令的技术可以很容易地扩展来呈现完整的某报纸、杂志或其它信息源。这一技术可以由某种定义好的分等级结构 (如某张报纸的体育版、商业版、城市版等) 加以补充, 以呈现这些信息。尽管如此, 该技术包括由确定用户提供的对象, 并用这些对象来有选择性地呈现信息。

[0081] 图 14A-14D 例示了用于呈现段落 800 的 SALT 的示例性 XML 代码。参照图 14A, 例示了一个头部部分 810 和数据部分 812。头部部分 810 包括初始化该文档的数据。数据部分 812 例示了由文本规格化器 / 分析器 604 分析及规格化后的段落 800。如所例示的那样, 多种标记符如 < 句子 >、< 实体名 = “股票指数” > 和 < 实体名 = “数字” > 已与段落 800 的各部分相关联。

[0082] 代码在图 14B 中接续, 在那里例示了多种语音应用语言标记符。例如, 该代码包括了一个听标记符 814、一个语法标记符 816 以及提示标记符 818 和 820。听标记符 814 初始化语音识别器, 并开始标识用户的话音输入内的对象。语法标记符 816 初始化语言模型, 在这一情况下该语言模型定义了导航规则 820 和查询规则 822。

[0083] 图 14C 例示了图 14B 中代码的接续。一个脚本标记符 826 标识了该代码的一个脚本部分的开头。该脚本部分包括多种操作数据呈现模块 16 的函数。一个执行命令函数 828 识别导航或查询命令, 并根据所识别出的对象调用必需的函数。一个初始化函数 830 开始从头播放段落 800。一个回移函数 832 和前移函数 834 被提供来分别回移一个句子和前移一个句子。在图 14D 中, 析取项函数 836 根据用户的话音输入从段落 800 中析取相关信息。显示屏 836 在如一个屏幕上显示段落 800。

[0084] 如上面所示的那样, 数据呈现模块 16 还可以用来呈现其它形式的未结构化文本。举例来说, 图 15 例示了一个语音邮件报文 840。对象 842 被用来呈现该报文的各部分。为了呈现该语音邮件报文 840 的各部分, 数据库接口 606 (图 8) 包括了一个将该语音邮件报文 840 转换到文本的语音识别器。在该报文被转换成文本后, 用文本规格化器 / 分析器 604 来标识该语音邮件报文的各相关部分。举例来说, 文本规格化器 / 分析器 604 可能标识出一个人、该报文的一个标题和 / 或数字如某个电话号码。与上面对段落 800 所描述的相类似, 根据这些标识来产生标记符。在呈现了该语音邮件报文或该语音邮件报文的一部分后, 用户可能请求重复相关部分。例如, 在图 15 的语音邮件报文中, 用户可能请求重复电话号码或该报文的标题。利用对象 842, 数据呈现模块 16 将这些对象与对应于语音邮件报文 840 中数据的标记符相关联。而后就能呈现所请求的数据了。

[0085] 在某一进一步实施例中, 可以处理多个语音邮件报文, 以使用导航对象提供对每条报文的有选择访问。利用报文中的信息 (即电话号码) 或通过访问其它信息如某个具有电话号码的人员列表, 可以用命令对象来指示回电等。在另一个关于图 11 中驾驶导向示例的例子中, 还可以从远端信息存储和 / 或远端应用中访问关于附近商业的信息。

[0086] 尽管本发明是参照特殊的实施例来描述的, 相关领域中的行家将认识到, 可以不背离本发明的精神和范围, 而在形式与细节上作出改变。

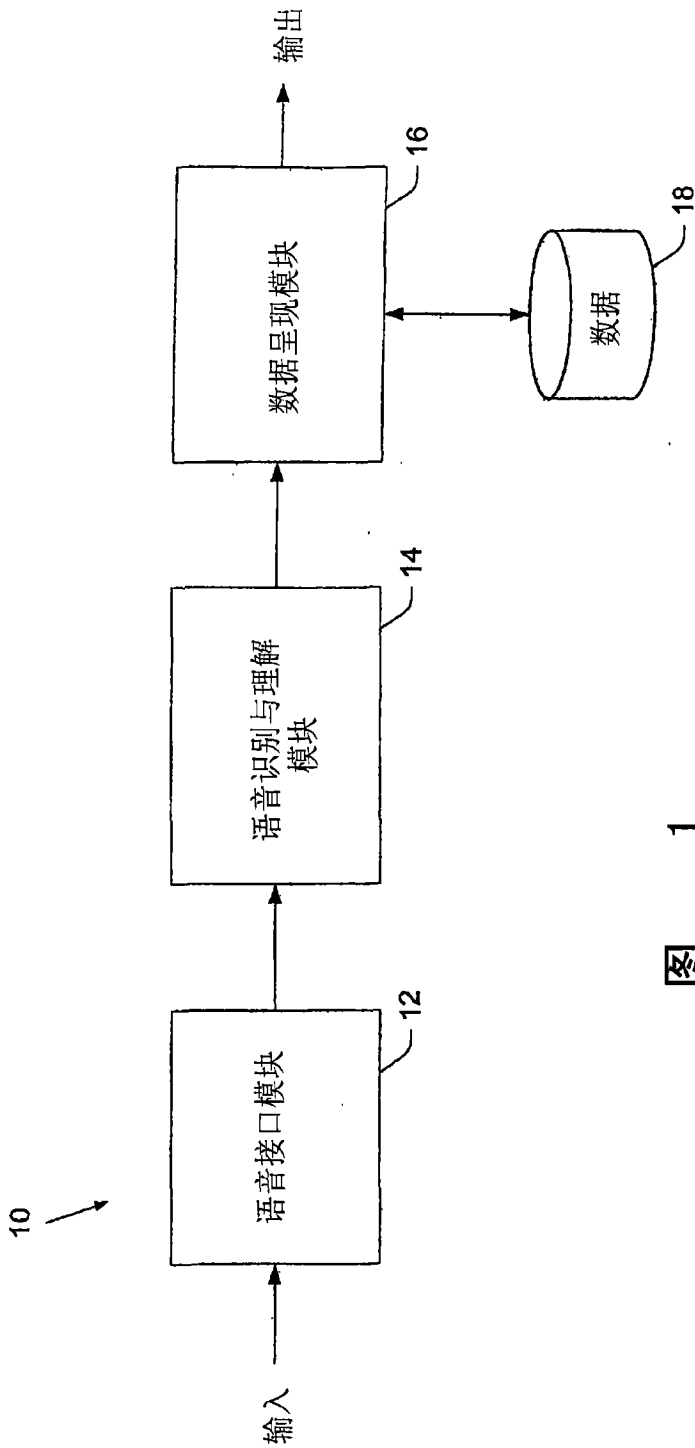
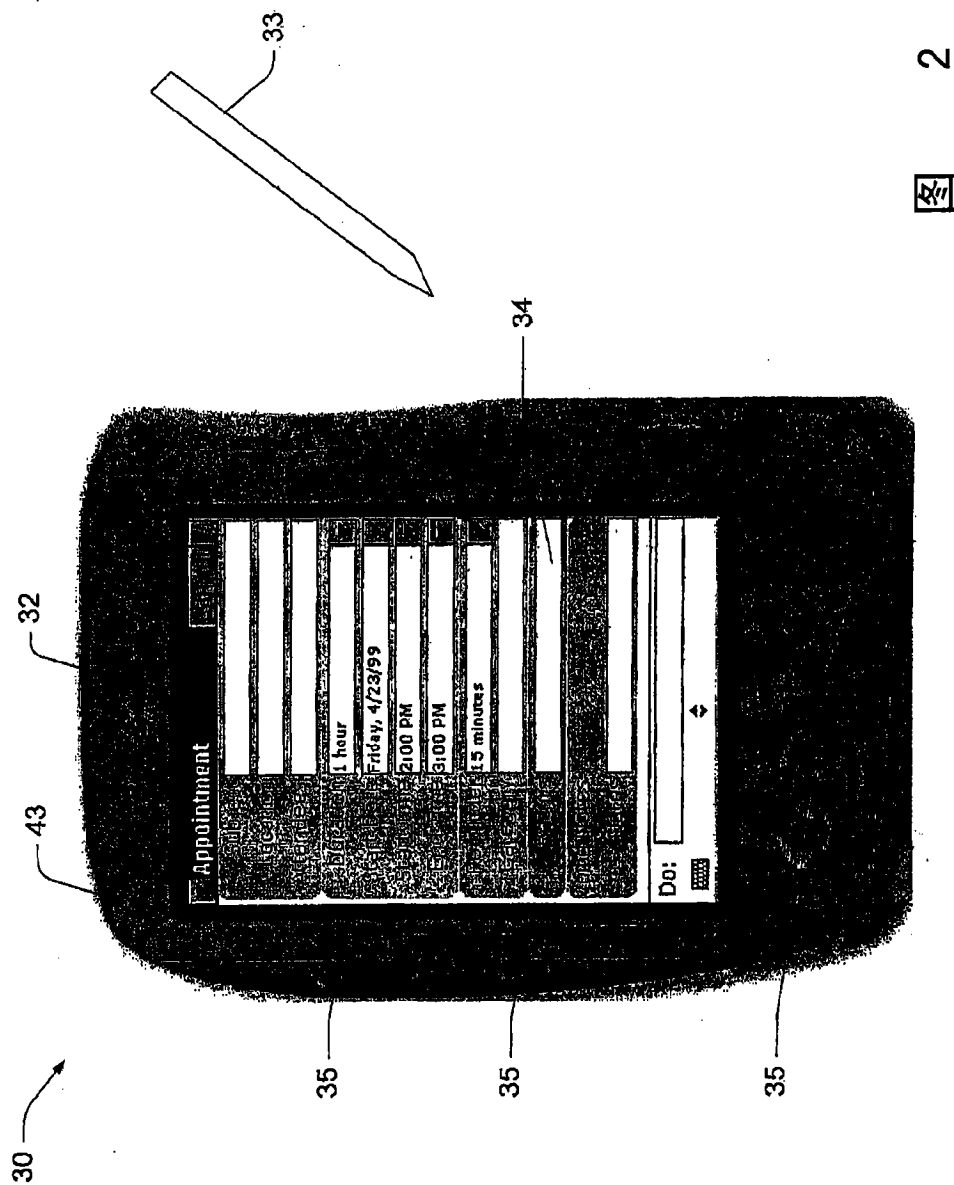


图 1



2



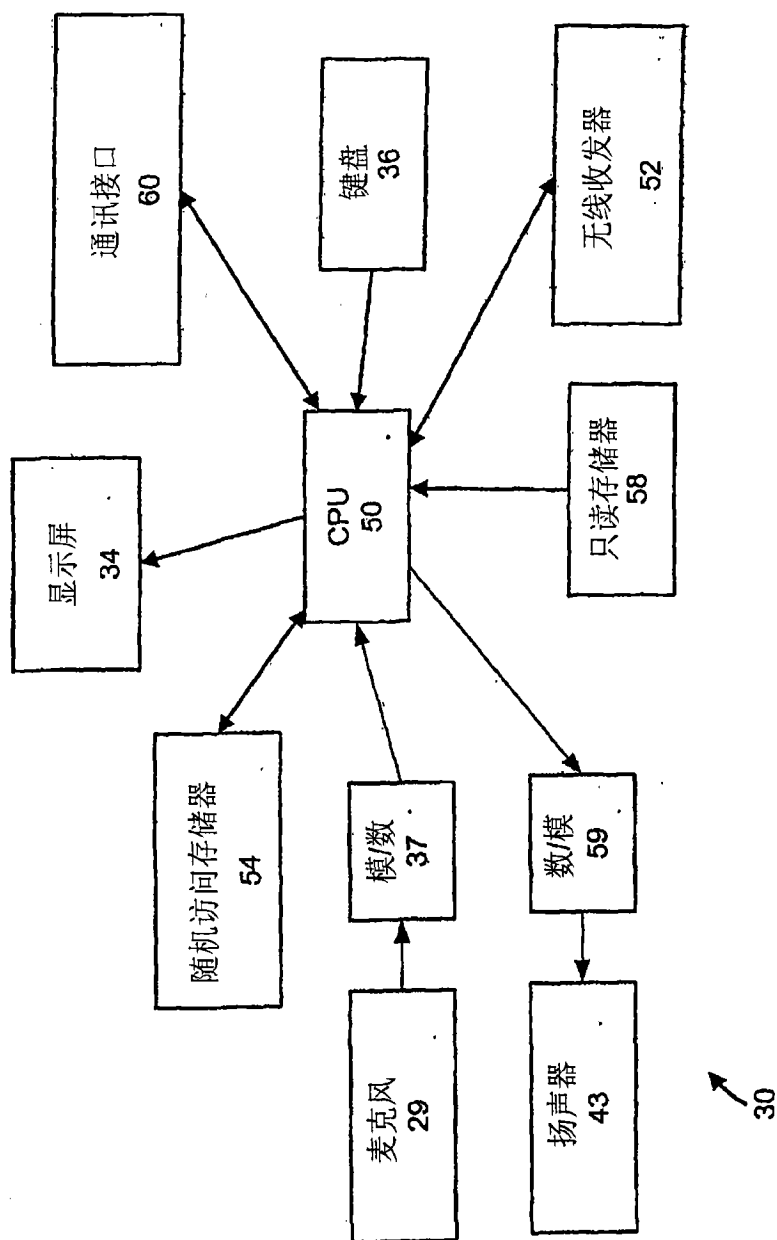


图 3

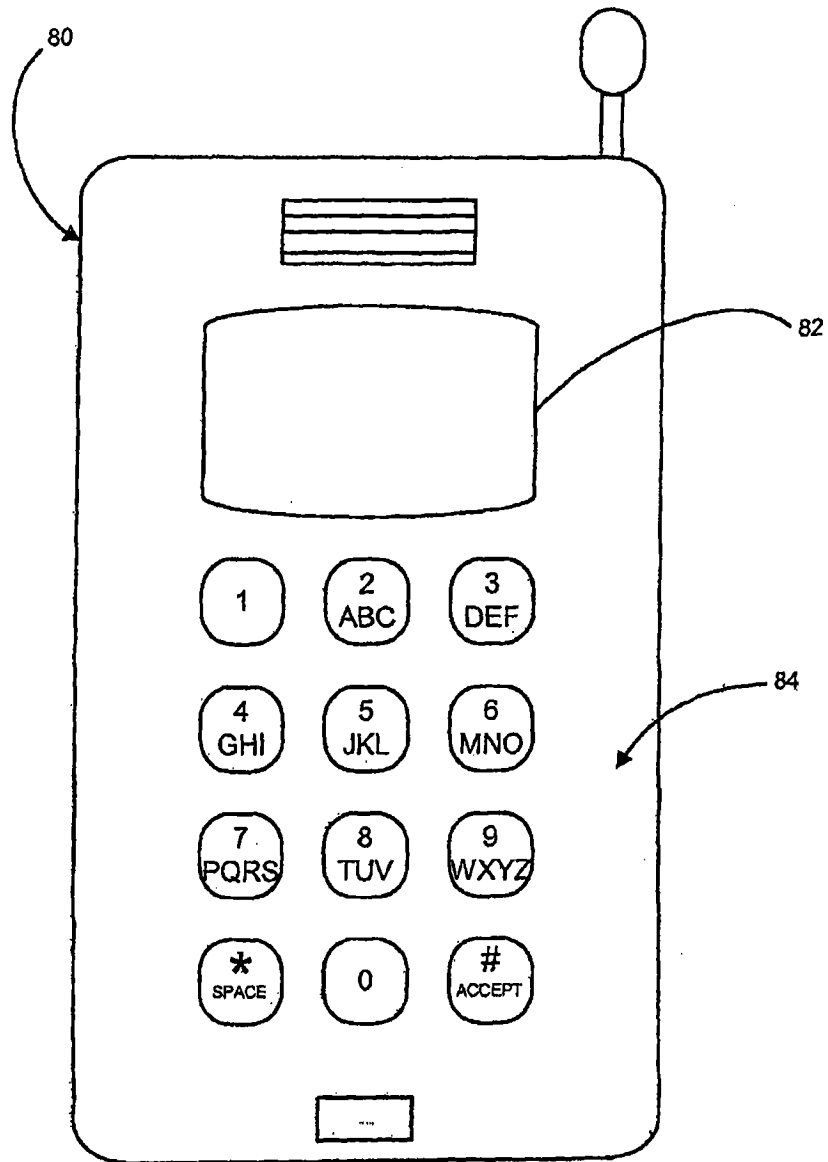


图 4

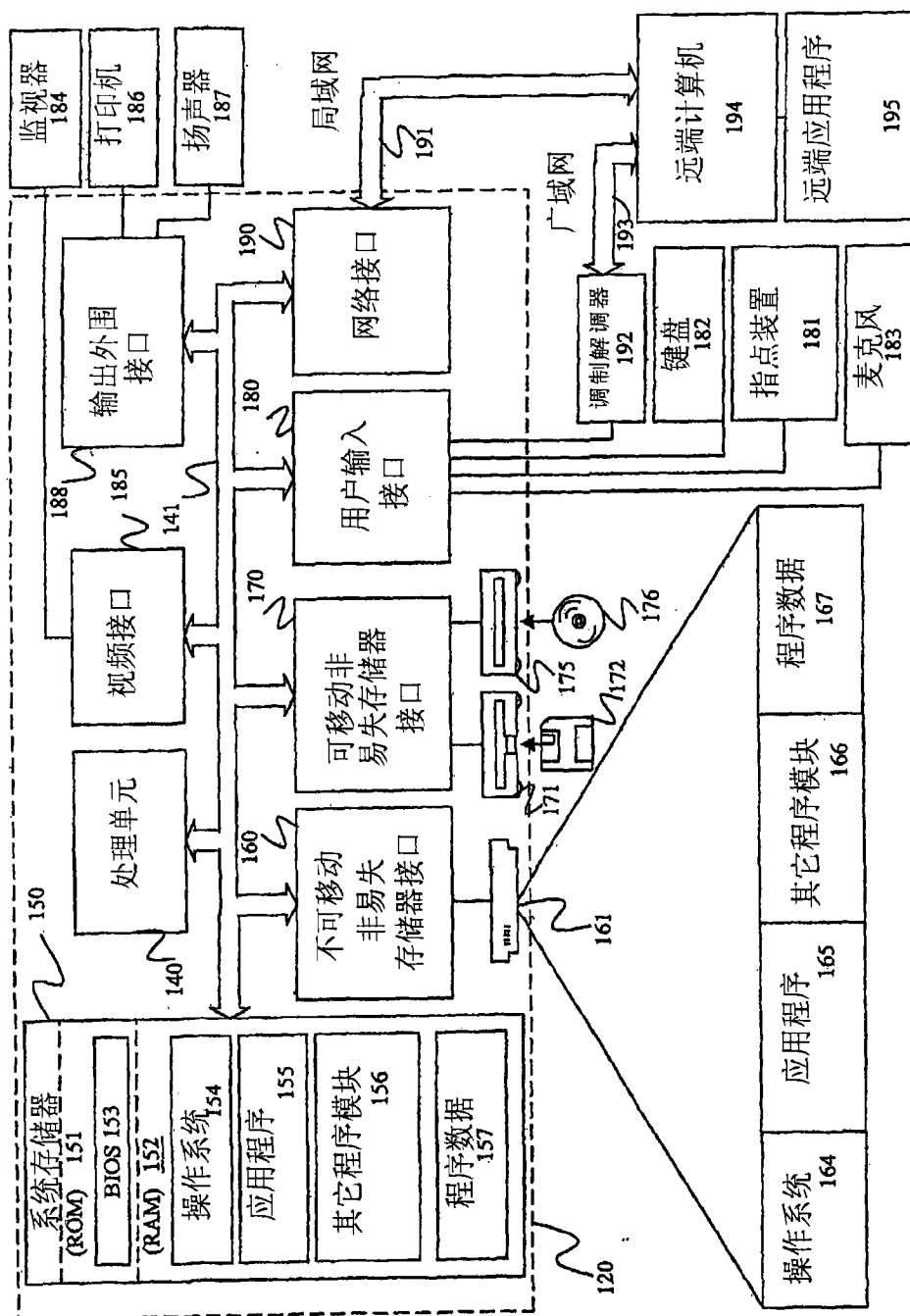


图 5

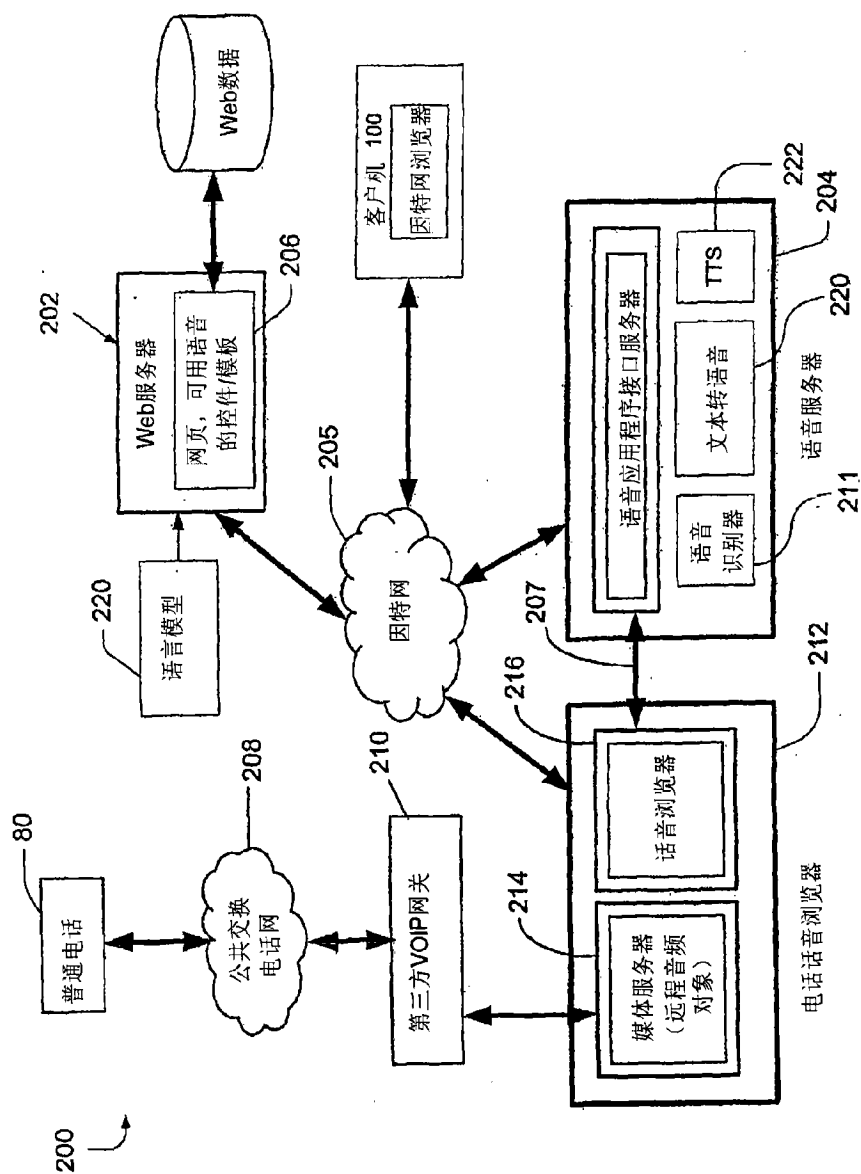


图 6

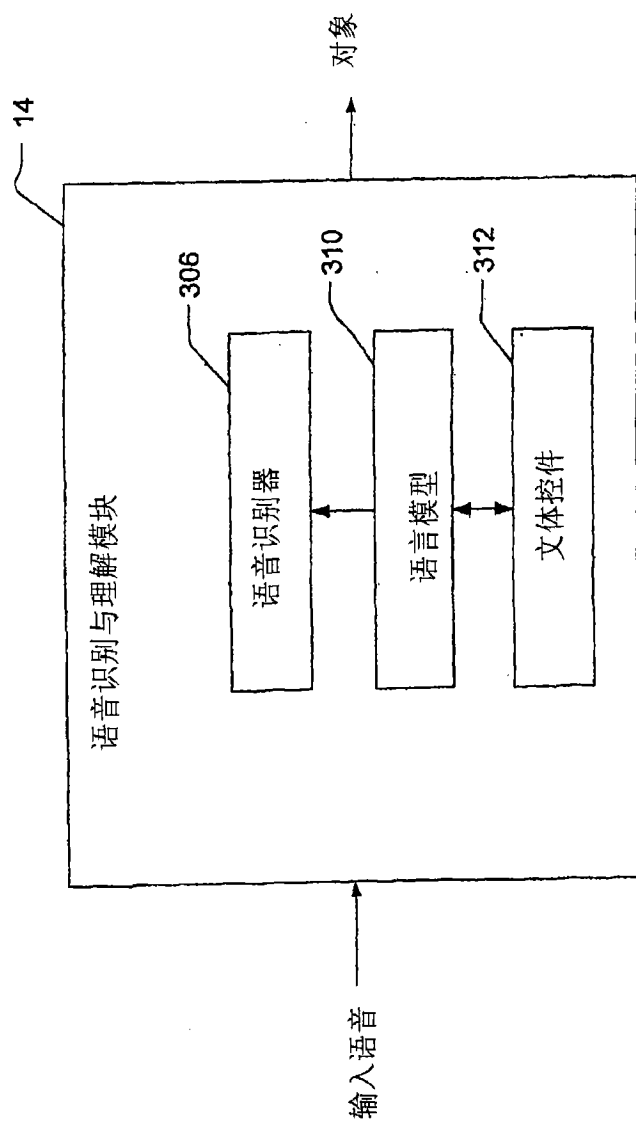


图 7

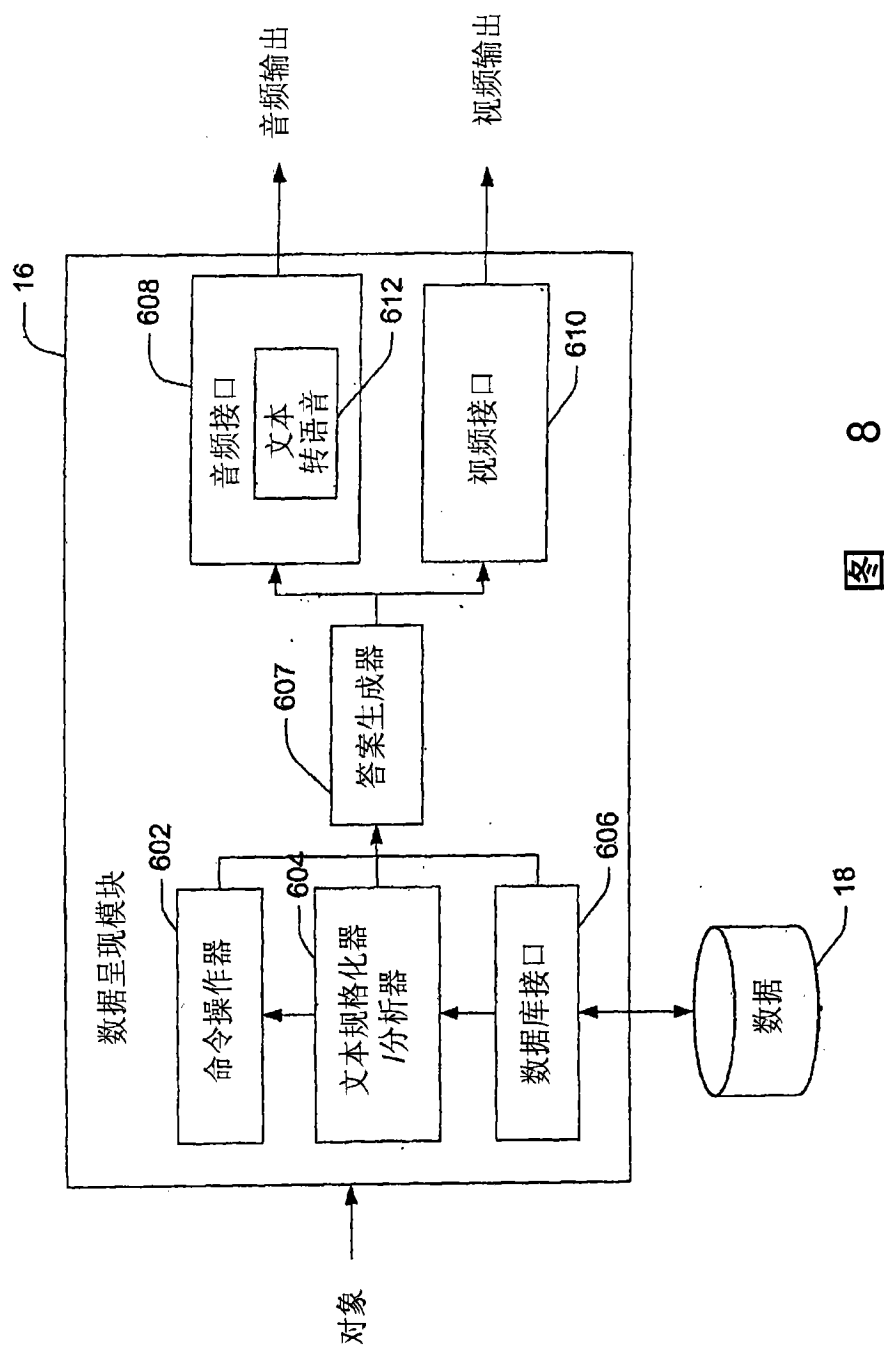


图 8

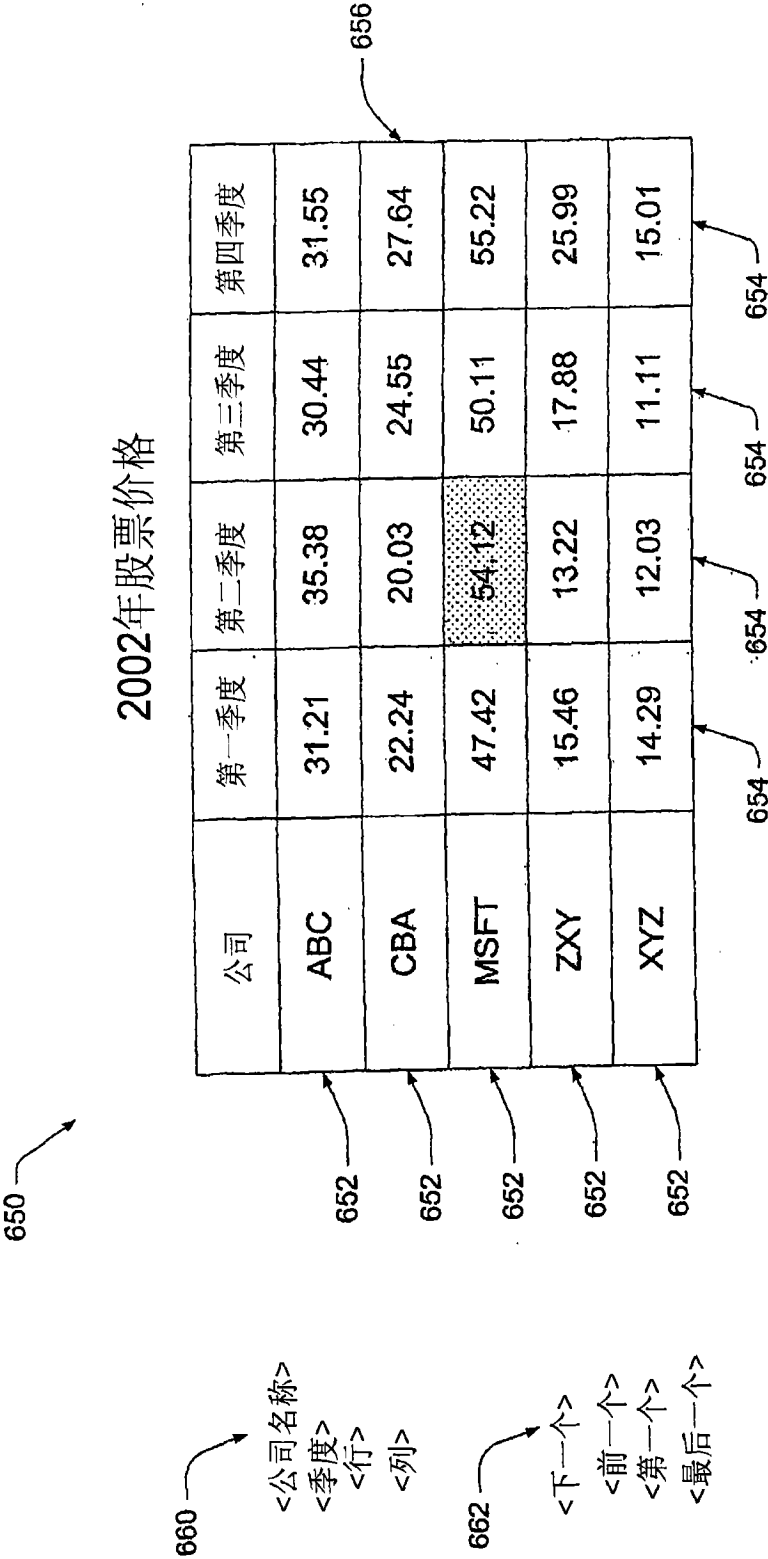


图 9

```

<html xmlns:salt="http://www.saltforum.org/2002/07">
<head>
<title>Simply SALT Demo</title>
<object id="MS-SALT" CLASSID="clsid:33cbfc53-a7de-491a-90f3-0e782a7e347a">
</object>
</head>
<?import namespace="salt" implementation="#MS-SALT">
<body onload="welcome.Start()">
  <xml id="database">
    <table>
      <company>
        <name>ABC</name>
        <q1>31.21</q1><q2>35.38</q2><q3>30.44</q3><q4>31.55</q4>
      </company>
      <company>
        <name>CBA</name>
        <q1>22.24</q1><q2>20.03</q2><q3>24.55</q3><q4>27.64</q4>
      </company>
      <company> ← 676
        <name>MSFT</name> ← 677
        <q1>47.42</q1><q2>54.12</q2><q3>50.11</q3><q4>55.22</q4>
      </company>
      <company> ← 678
        <name>ZXY</name>
        <q1>15.46</q1><q2>13.22</q2><q3>17.88</q3><q4>25.99</q4>
      </company>
      <company>
        <name>XYZ</name>
        <q1>14.29</q1><q2>12.03</q2><q3>11.11</q3><q4>15.01</q4>
      </company>
    </table>
  </xml>

  <input id="querySubject">
  <input id="queryObject">
  <input id="queryResult">

```

670

672

674

图 10A

```

<salt:listen id="command" ← 680
  onReco="doQuery()"
  onError="alert('Error code='+ event.srcElement.status)"
  <salt:grammar id="mainGrammar" ← 682
    <grammar langid="409">
      <rule name="start" toplevel="active">
        <phrase>what is ?the <ruleref name="object"/> result of <ruleref name="subject"/></phrase>
      </rule>
      <rule name="subject">
        <list proptype="subject">
          <phrase valstr="ABC">A B C </phrase>
          <phrase valstr="CBA">C B A </phrase>
          <phrase valstr="MSFT">?the Microsoft ?Corporation</phrase>
          <phrase valstr="MSFT">M. S. F. T. </phrase>
          <phrase valstr="ZXY">Z. X. Y. </phrase>
          <phrase valstr="XYZ">X. Y. Z. </phrase>
        </list>
      </rule>
      <rule name="object">
        <list proptype="object">
          <phrase valstr="q1"> first quarter</phrase>
          <phrase valstr="q1"> Q. one</phrase>
          <phrase valstr="q2"> second quarter</phrase>
          <phrase valstr="q2"> Q. two </phrase>
          <phrase valstr="q3"> third quarter </phrase>
          <phrase valstr="q3"> Q. three< /phrase>
          <phrase valstr="q4"> fourth quarter </phrase>
          <phrase valstr="q4"> Q four </phrase>
        </list>
      </rule>
    </grammar>
  </salt:grammar>
</salt:listen>

<salt:prompt id="tts">The <salt:value targetElement="queryObject"/> of ← 684
  <salt:value targetElement="querySubject"/> is
  <salt:value targetElement="queryResult"/>
</salt:prompt>

<salt:prompt id="welcome" oncomplete="command Start()">Please say your query.</salt:prompt> ← 686
</body>

```

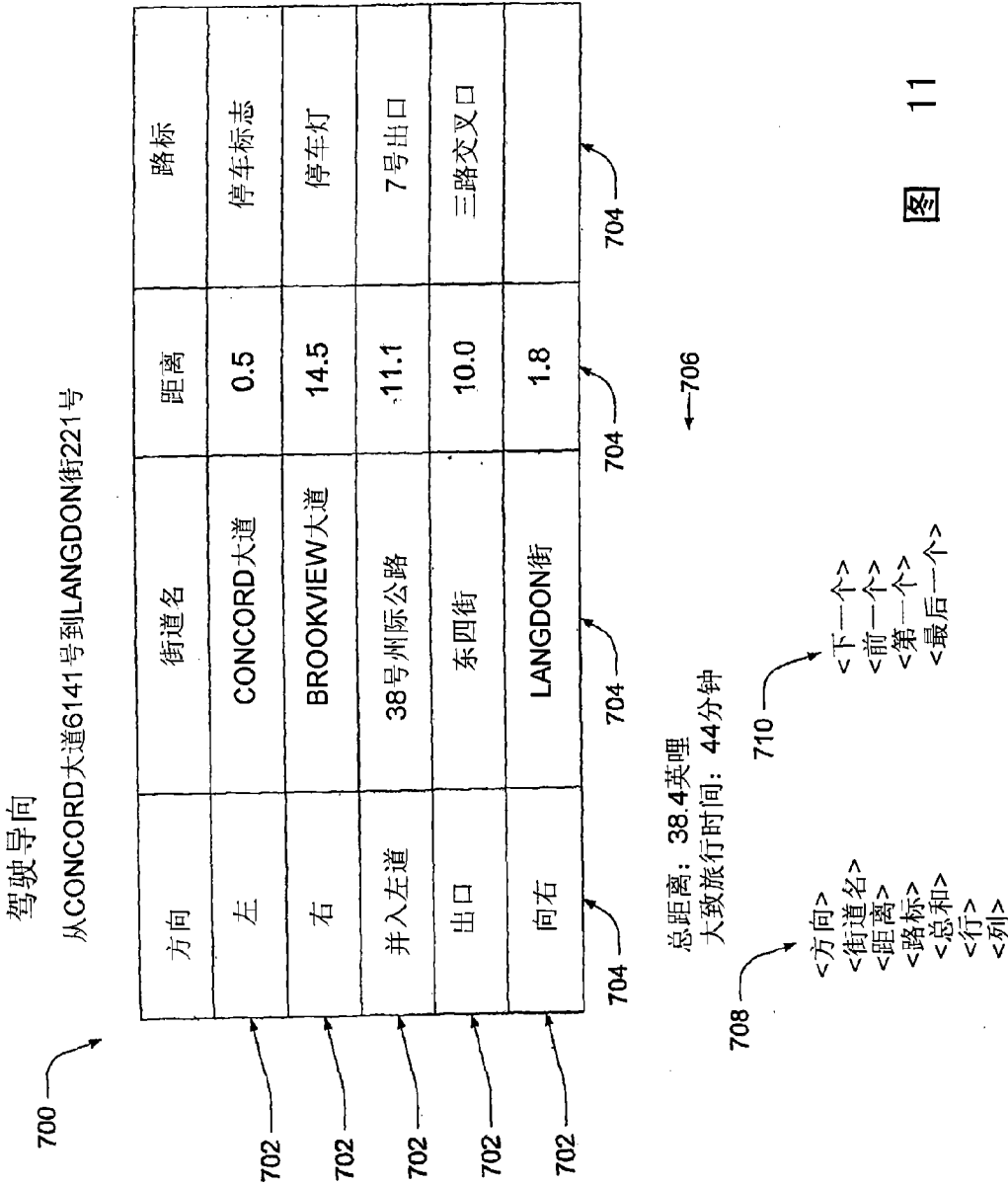
图 10B

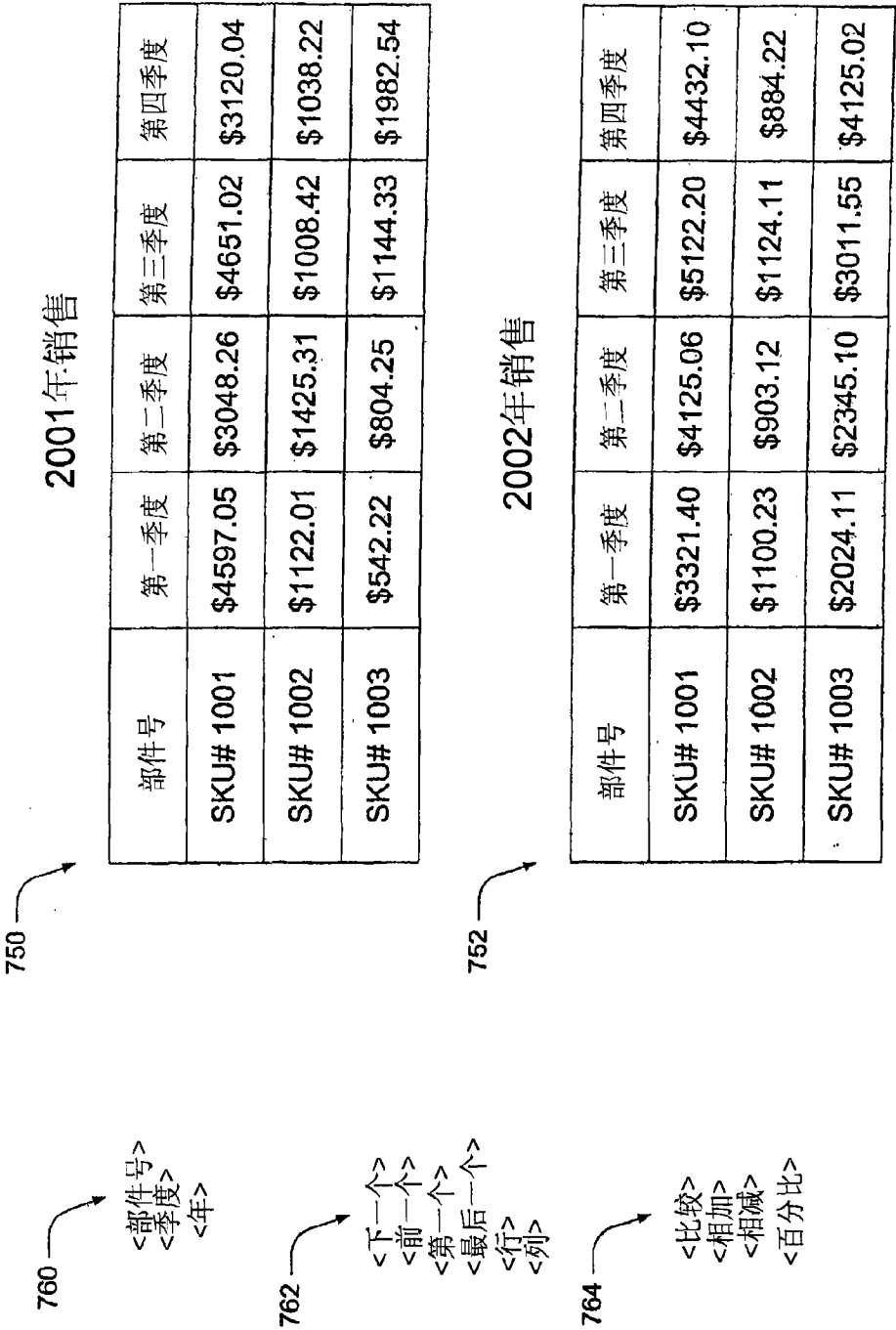
```
<script>
function doQuery() {
    var obj = command.recoresult.selectSingleNode("/sml/object");
    var sub = command.recoresult.selectSingleNode("/sml/subject");
    // compose query string to search for the data
    var queryString = "company/name='"+ sub.text + "' and company/"
+ obj.text;
    queryResult.value = database.selectSingleNode(queryString).text;
    // compose answer string with user's own wording
    queryObject.value = obj.attributes.selectSingleNode("text").value;
    querySubject.value =
sub.attributes.selectSingleNode("text").value;
    // activate TTS
    tts.Start();
}
</script>
</html>
```



692

图 10C





图

12

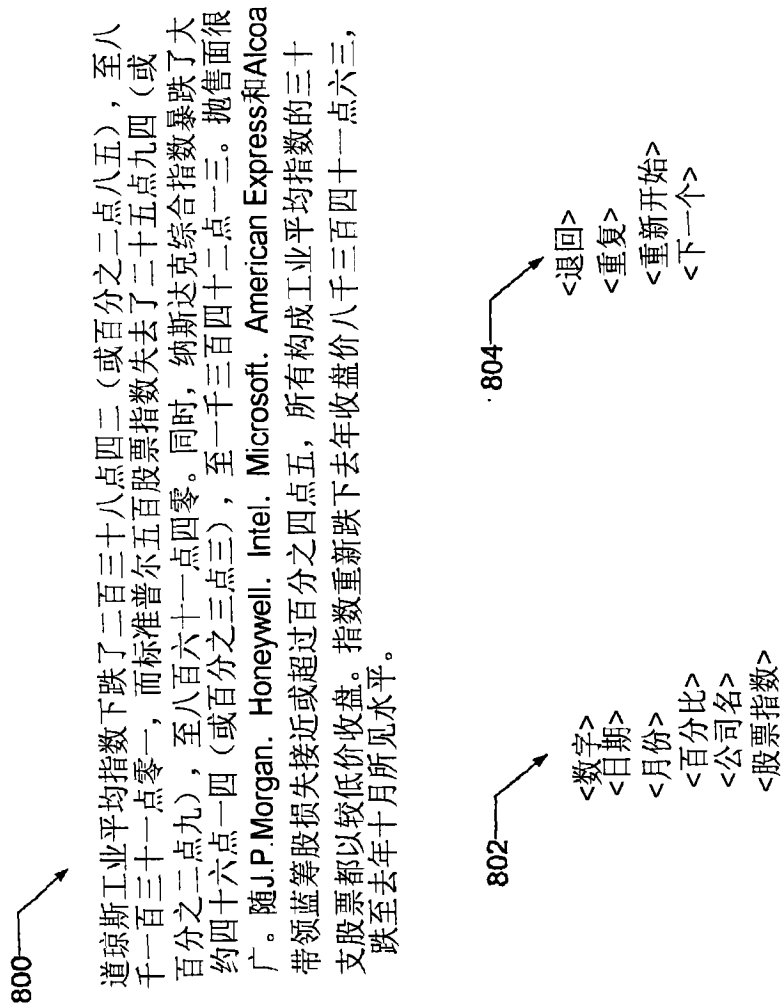


图 13

```

<html xmlns:salt="http://www.saltforum.org/2002/07">
<head>
<title>News Reader Demo</title>
<object id="MS-SALT" CLASSID="clsid:33cbfc53-a7de-491a-90f3-0e782a7e347a">
</object>
</head>
<?import namespace="salt" implementation="#MS-SALT">
<body onload="init()">

```

810

```

<!-- The following XML is the outcome of passing WSJ article through a text analyzer that tags
name
      entities and normalizes text for prosody assignments.
-->
<xml id="news">
<metadata>
<sentence>The <entity name="stock index">Dow Jones Industrial Average </entity>tumbled
<entity name="number">two hundred thirty eight point four two, </entity>or <entity
name="percent">two point eight five percent, </entity>to <entity name="number">eight
thousand one hundred thirty one point zero one, </entity>while the <entity name="stock
index">Standard and Poor's five hundred stock index </entity>lost <entity
name="number">twenty five point nine four, </entity>or <entity name="percent">two point nine
percent. </entity>to <entity name="number">eight hundred sixty one point four zero. </
entity></sentence>
<sentence>Meanwhile, the <entity name="stock index">Nasdaq Composite Index </
entity>slumped about <entity name="number">fourty six point one four, </entity>or <entity
name="percent">three point three percent, </entity>to <entity name="number">one thousand
three hundred forty two point one three</entity>. </sentence>
<sentence>The selling was broad based. </sentence>
<sentence>All <entity name="number">thirty </entity>stocks that make up the industrial
average closed lower, with <entity name="company name">J. P. Morgan, </entity><entity
name="company name">Hoenywell, </entity><entity name="company name">Intel, </
entity><entity name="company name">Microsoft, </entity><entity name="company
name">American Express, </entity>and <entity name="company name">Alcoa </
entity>leading the blue chips with losses near or above <entity name="percent">four point five
percent. </entity></sentence>
<sentence>The index again fell below last year's closing point of <entity
name="number">eight thousand three hundred forty one point six three, </entity>dropping to
levels last seen in late <entity name="month">October. </entity></sentence>
</metadata>
</xml>

```

812

```

<div id="debug"></div> <!-- display the text being played here -->

```

图 14A


```

<salt:listen id="reco" ← 814
  onReco="execute_cmd()" onNoReco="reprompt.Start()" onsilence="move_next()"
  <salt:grammar id="name" ← 816
    <grammar langid="409">
      <rule name="cmds" toplevel="active">
        < >
          <ruleref name="navigate"/>
          <ruleref name="query"/>
        </ >
      </rule>
      <rule name="navigate">
        <l proptype="navigate">
          <p valstr="back">go back <o>one sentence</o></p>
          <p valstr="repeat"><l> <p>replay</p><p>repeat</p></l> <o>?the sentence</o> </p>
          <p valstr="restart"><l><p>restart</p><p>repeat from ?the beginning</p></l></p>
          <p valstr="next"><l><p>next ?sentence</p><p>continue</p></l></p>
        </l>
      </rule>
      <rule name="query">
        <p>
          <l><p>repeat</p><p>say ?the</p></l>
          <l proptype="query">
            <p valstr="number"> ?the number </p>
            <p valstr="date"> ?the date </p>
            <p valstr="month"> ?the month </p>
            <p valstr="percent"> ?the percentage </p>
            <p valstr="company name"> ?the company names </p>
            <p valstr="stock index"> ?the stock index </p>
          </l>
          <o>again</o>
        </p>
      </rule>
    </grammar>
  </salt:grammar>
</salt:listen>

<salt:prompt id="reprompt" onComplete="reco.Start()" ← 818
  Sorry, I missed that. Please say the command again.
</salt:prompt>

<salt:prompt id="fts" onComplete="reco.Start()" ← 820

```

822 {

824 {

图 14B

```

</body>

<script>
    var curSent, curPrompt;
    function execute_cmd() {
        // the event handler for recognizing navigation or query commands
        var node = reco.recoresult.firstChild;
        if (node.nodeName == "query") {
            extract_item(node.text);
        } else if (node.nodeName == "navigate") {
            if (node.text == "back")
                move_back();
            else if (node.text == "repeat")
                display();
            else if (node.text == "restart")
                init();
            else move_next();
        } else
            move_next();
    }
    function init() { // event handler to play the text from the very beginning
        curSent = news.XMLDocument.selectSingleNode("metadata/sentence");
        curPrompt = curSent.text;
        display();
    }
    function move_back() { // move back one sentence
        if (curSent.previousSibling != null) {
            curSent = curSent.previousSibling;
            curPrompt = curSent.text;
        }
        display();
    }
    function move_next() { // move on to the next sentence
        if (curSent.nextSibling != null) {
            curSent = curSent.nextSibling;
            curPrompt = curSent.text;
        }
        display();
    }

```

828

830

832

834

图 14C

```
function extract_item(target) { // processing query items
    var str = "/entity[@name=" + target + " | mentioned ";
    var candidates = curSent.selectNodes(str);
    var count = candidates.length;
    if (count == 0)
        str += "nothing.";
    else {
        for (var i = 0; i < count; ++i)
            str += candidates.item(i).text + " ";
    }
    curPrompt = str;
    display();
}
function display() {
    debug.innerText = curPrompt;
    tts.Start(curPrompt);
}
</script>
```

836

838

图 14D

840

"喂Mel，我是Todd。我找你是关于我们的旅行。请给我电话612-333-8888。"

842

<人>
<标题>
<电话号码>

图 15