



(12) 发明专利申请

(10) 申请公布号 CN 103927383 A

(43) 申请公布日 2014. 07. 16

(21) 申请号 201410177926. 8

(22) 申请日 2014. 04. 29

(71) 申请人 车智互联(北京)科技有限公司
地址 100080 北京市海淀区丹棱街3号B座
11层1110、1111室

(72) 发明人 安凯

(74) 专利代理机构 北京思睿峰知识产权代理有限公司 11396
代理人 靳春鹰 赵爱军

(51) Int. Cl.
G06F 17/30(2006. 01)
H04L 29/08(2006. 01)

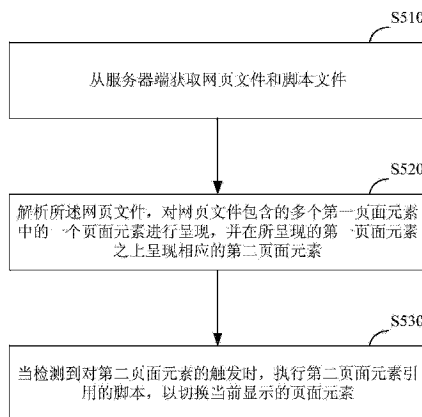
权利要求书2页 说明书18页 附图3页

(54) 发明名称

一种网页呈现设备和方法

(57) 摘要

本发明公开一种网页呈现设备和方法,属于互联网技术领域。网页呈现方法包括:服务器端根据包括多个数据项目的数据内容构造网页文件和脚本文件,网页文件包括多个第一页面元素和多个第二页面元素,每个第一页面元素包括数据内容中的一个数据项目,每个第二页面元素包括对脚本文件中脚本的引用;浏览器解析网页文件,对多个第一页面元素中的一个页面元素进行呈现,并在所呈现的第一页面元素之上呈现相应的第二页面元素,当检测到对第二页面元素的触发时,执行第二页面元素引用的脚本,以根据各数据项目之间的逻辑关系切换当前显示的页面元素。本发明实现了在浏览器中以分页方式显示数据内容,并提高了浏览器的网页显示速度,降低了服务器端的负载。



1. 一种网页呈现方法,适于在浏览器中执行,所述方法包括:

从服务器端获取网页文件和脚本文件,所述网页文件和脚本文件由服务器端根据包括多个数据项目的数据内容构造得到,其中所构造的网页文件包括多个第一页面元素和多个第二页面元素,每个第一页面元素包括数据内容中的一个数据项目,每个第二页面元素包括对脚本文件中脚本的引用,所引用的脚本适于根据各数据项目之间的逻辑关系切换当前显示的页面元素;

解析所述网页文件,对所述多个第一页面元素中的一个页面元素进行呈现,并在所呈现的第一页面元素之上呈现相应的第二页面元素;以及

当检测到对第二页面元素的触发时,执行第二页面元素引用的脚本,以隐藏当前显示的第一页面元素和第二页面元素,并呈现与当前隐藏的第一页面元素相关联的第一页面元素和对应的第二页面元素。

2. 如权利要求 1 所述的网页呈现方法,其中,所述第一页面元素和第二页面元素为 <div> 标签,且所述脚本通过更改 <div> 标签的" display" 属性来切换当前显示的页面元素。

3. 如权利要求 2 所述的网页呈现方法,其中,所述数据内容包括第一数据条目列表和第二数据条目列表,且每个数据项目包括第一数据条目列表中的一个数据条目和第二数据条目列表中与该第一数据条目对应的第二数据条目。

4. 如权利要求 3 所述的网页呈现方法,其中,所述第一页面元素对应的 <div> 标签为嵌套结构,其中外层 <div> 标签的内容为数据内容所在的页码,内层 <div> 标签的内容为第一数据条目以及与该第一数据条目对应的第二数据条目,所述第二页面元素对应的 <div> 标签的内容为按钮,所述按钮引用的脚本适于根据对第一数据条目对应的第二数据条目的选择结果切换当前显示的页面元素。

5. 一种网页呈现方法,适于在服务器端执行,所述方法包括:

根据包括多个数据项目的数据内容构造网页文件和脚本文件,其中所构造的网页文件包括多个第一页面元素和多个第二页面元素,每个第一页面元素包括数据内容中的一个数据项目,每个第二页面元素包括对脚本文件中脚本的引用,所引用的脚本适于根据各数据项目之间的逻辑关系切换当前显示的页面元素;

根据浏览器的请求将所构造的网页文件和脚本文件返回给浏览器,以供浏览器解析所述网页文件,对所述多个第一页面元素中的一个页面元素进行呈现,并在所呈现的第一页面元素之上呈现相应的第二页面元素,以及,当检测到对第二页面元素的触发时,执行第二页面元素引用的脚本,以隐藏当前显示的第一页面元素和第二页面元素,并呈现与当前隐藏的第一页面元素相关联的第一页面元素和对应的第二页面元素。

6. 如权利要求 5 所述的网页呈现方法,其中,所述第一页面元素和第二页面元素为 <div> 标签,且所述脚本通过更改 <div> 标签的" display" 属性来切换当前显示的页面元素。

7. 如权利要求 6 所述的网页呈现方法,其中,所述数据内容包括第一数据条目列表和第二数据条目列表,且每个数据项目包括第一数据条目列表中的一个数据条目和第二数据条目列表中与该第一数据条目对应的第二数据条目。

8. 一种网页呈现设备,驻留在浏览器中,所述设备包括:

请求单元,适于从服务器端获取网页文件和脚本文件,所述网页文件和脚本文件由服务器端根据包括多个数据项目的数据内容构造得到,其中所构造的网页文件包括多个第一页面元素和多个第二页面元素,每个第一页面元素包括数据内容中的一个数据项目,每个第二页面元素包括对脚本文件中脚本的引用,所引用的脚本适于根据各数据项目之间的逻辑关系切换当前显示的页面元素;

解析单元,适于解析所述网页文件,对所述多个第一页面元素中的一个页面元素进行呈现,并在所呈现的第一页面元素之上呈现相应的第二页面元素;以及

处理单元,适于当检测到对第二页面元素的触发时,执行第二页面元素引用的脚本,以隐藏当前显示的第一页面元素和第二页面元素,并呈现与当前隐藏的第一页面元素相关联的第一页面元素和对应的第二页面元素。

9. 一种网页呈现设备,驻留在服务器端,所述设备包括:

构造单元,适于根据包括多个数据项目的数据内容构造网页文件和脚本文件,其中所构造的网页文件包括多个第一页面元素和多个第二页面元素,每个第一页面元素包括数据内容中的一个数据项目,每个第二页面元素包括对脚本文件中脚本的引用,所引用的脚本适于根据各数据项目之间的逻辑关系切换当前显示的页面元素;

响应单元,适于根据浏览器的请求将所构造的网页文件和脚本文件返回给浏览器,以供浏览器解析所述网页文件,对所述多个第一页面元素中的一个页面元素进行呈现,并在所呈现的第一页面元素之上呈现相应的第二页面元素,以及,当检测到对第二页面元素的触发时,执行第二页面元素引用的脚本,以隐藏当前显示的第一页面元素和第二页面元素,并呈现与当前隐藏的第一页面元素相关联的第一页面元素和对应的第二页面元素。

10. 一种网页呈现系统,所述系统包括浏览器和服务端,所述浏览器与所述服务端通信连接,且所述浏览器包括如权利要求8所述的网页呈现设备,所述服务端包括如9所述的网页呈现设备。

一种网页呈现设备和方法

技术领域

[0001] 本发明涉及互联网技术领域,具体涉及一种网页呈现设备和方法。

背景技术

[0002] 中国互联网信息中心 (China Internet Network Information Center, CNNIC) 调查报告显示,截至到 2013 年 6 月底,我国网民规模达 5.91 亿。而随着互联网的快速发展,网络成为人们购物、获取资讯和沟通交流的第一渠道,一些传统的商业也快速迁移到互联网上。

[0003] 企业为了更好地收集用户反馈信息,需要发布调查问卷。传统调查问卷需要耗费大量的人力、时间和财力,对问卷结果的统计也会有偏差及延迟,因此迫切需要一款能够满足需求的在线调查问卷系统。

[0004] 如图 6 所示,现有的在线调查问卷的一种实现方式为,将调查问卷的所有试题在一个网页中呈现出来,其缺点在于:没有对试题进行分页呈现,缺少与用户之间的交互性;没有逻辑跳转,一些选择性显示的试题无法展现,例如:不能根据男性用户展现符合男性的试题。

[0005] 现有的在线调查问卷的另一种实现方式为,通过动态请求服务器实现试题的分页,即每次在网页上呈现一个试题,在获取到用户对该试题的答案之后,从服务器端下载下一个试题后进行呈现。该种实现方式虽然提供了对试题的分页,但是每显示一页,都需要与服务器端进行交互,例如发送 http 请求到服务器端,根据服务器端的响应进行当前内容的显示。由于存在与服务器端的多重交互,一方面降低了网页呈现的速度,页面打开时间变长,另一方面也增加了服务器端(网站)的负载。

发明内容

[0006] 鉴于上述问题,提出了本发明以便提供一种克服上述问题或者至少部分地解决上述问题的网页呈现设备和方法。

[0007] 根据本发明的一个方面,提供了一种网页呈现方法,适于在浏览器中执行,所述方法包括:

[0008] 从服务器端获取网页文件和脚本文件,所述网页文件和脚本文件由服务器端根据包括多个数据项目的数据内容构造得到,其中所构造的网页文件包括多个第一页面元素和多个第二页面元素,每个第一页面元素包括数据内容中的一个数据项目,每个第二页面元素包括对脚本文件中脚本的引用,所引用的脚本适于根据各数据项目之间的逻辑关系切换当前显示的页面元素;

[0009] 解析所述网页文件,对所述多个第一页面元素中的一个页面元素进行呈现,并在所呈现的第一页面元素之上呈现相应的第二页面元素;以及

[0010] 当检测到对第二页面元素的触发时,执行第二页面元素引用的脚本,以隐藏当前显示的第一页面元素和第二页面元素,并呈现与当前隐藏的第一页面元素相关联的第一页

面元素和对应的第二页面元素。

[0011] 根据本发明的另一方面,提供了一种网页呈现方法,适于在服务器端执行,所述方法包括:

[0012] 根据包括多个数据项目的数据内容构造网页文件和脚本文件,其中所构造的网页文件包括多个第一页面元素和多个第二页面元素,每个第一页面元素包括数据内容中的一个数据项目,每个第二页面元素包括对脚本文件中脚本的引用,所引用的脚本适于根据各数据项目之间的逻辑关系切换当前显示的页面元素;

[0013] 根据浏览器的请求将所构造的网页文件和脚本文件返回给浏览器,以供浏览器解析所述网页文件,对所述多个第一页面元素中的一个页面元素进行呈现,并在所呈现的第一页面元素之上呈现相应的第二页面元素,以及,当检测到对第二页面元素的触发时,执行第二页面元素引用的脚本,以隐藏当前显示的第一页面元素和第二页面元素,并呈现与当前隐藏的第一页面元素相关联的第一页面元素和对应的第二页面元素。

[0014] 根据本发明的又一方面,提供了一种网页呈现设备,驻留在浏览器中,所述设备包括:

[0015] 请求单元,适于从服务器端获取网页文件和脚本文件,所述网页文件和脚本文件由服务器端根据包括多个数据项目的数据内容构造得到,其中所构造的网页文件包括多个第一页面元素和多个第二页面元素,每个第一页面元素包括数据内容中的一个数据项目,每个第二页面元素包括对脚本文件中脚本的引用,所引用的脚本适于根据各数据项目之间的逻辑关系切换当前显示的页面元素;

[0016] 解析单元,适于解析所述网页文件,对所述多个第一页面元素中的一个页面元素进行呈现,并在所呈现的第一页面元素之上呈现相应的第二页面元素;以及

[0017] 处理单元,适于当检测到对第二页面元素的触发时,执行第二页面元素引用的脚本,以隐藏当前显示的第一页面元素和第二页面元素,并呈现与当前隐藏的第一页面元素相关联的第一页面元素和对应的第二页面元素。

[0018] 根据本发明的又一方面,提供了一种网页呈现设备,驻留在服务器端,所述设备包括:

[0019] 构造单元,适于根据包括多个数据项目的数据内容构造网页文件和脚本文件,其中所构造的网页文件包括多个第一页面元素和多个第二页面元素,每个第一页面元素包括数据内容中的一个数据项目,每个第二页面元素包括对脚本文件中脚本的引用,所引用的脚本适于根据各数据项目之间的逻辑关系切换当前显示的页面元素;

[0020] 响应单元,适于根据浏览器的请求将所构造的网页文件和脚本文件返回给浏览器,以供浏览器解析所述网页文件,对所述多个第一页面元素中的一个页面元素进行呈现,并在所呈现的第一页面元素之上呈现相应的第二页面元素,以及,当检测到对第二页面元素的触发时,执行第二页面元素引用的脚本,以隐藏当前显示的第一页面元素和第二页面元素,并呈现与当前隐藏的第一页面元素相关联的第一页面元素和对应的第二页面元素

[0021] 可选地,所述第一页面元素和第二页面元素为 <div> 标签,且所述脚本通过更改 <div> 标签的" display" 属性来切换当前显示的页面元素。

[0022] 可选地,所述数据内容包括第一数据条目列表和第二数据条目列表,且每个数据项目包括第一数据条目列表中的一个数据条目和第二数据条目列表中与该第一数据条目

对应的第二数据条目。

[0023] 可选地,所述第一页面元素对应的<div>标签为嵌套结构,其中外层<div>标签的内容为数据内容所在的页码,内层<div>标签的内容为第一数据条目以及与该第一数据条目对应的第二数据条目,所述第二页面元素对应的<div>标签的内容为按钮,所述按钮引用的脚本适于根据对第一数据条目对应的第二数据条目的选择结果切换当前显示的页面元素。

[0024] 根据本发明的又一方面,提供了一种网页呈现系统,所述系统包括浏览器和服务端,所述浏览器与所述服务器端通信连接,且所述浏览器包括驻留在其中的上述网页呈现设备,所述服务器端包括驻留在其中的上述网页呈现设备。

[0025] 根据本发明上述的一个或多个技术方案,通过一次生成静态网页文件(例如html文件),数据内容在浏览器端一次下载完成,并通过在浏览器端执行脚本来根据数据内容中各数据项目之间的逻辑关系实现数据内容的分页呈现,由于不需要与服务器端进行多重交互,因而提高了浏览器的网页显示速度,并降低了服务器端的负载。

[0026] 上述说明仅是本发明技术方案的概述,为了能够更清楚了解本发明的技术手段,而可依照说明书的内容予以实施,并且为了让本发明的上述和其它目的、特征和优点能够更明显易懂,以下特举本发明的具体实施方式。

附图说明

[0027] 通过阅读下文优选实施方式的详细描述,各种其他的优点和益处对于本领域普通技术人员将变得清楚明了。附图仅用于示出优选实施方式的目的,而并不认为是对本发明的限制。而且在整个附图中,用相同的参考符号表示相同的部件。在附图中:

[0028] 图1示出了根据本发明一个实施例的网页呈现系统的结构图;

[0029] 图2示出了根据本发明一个实施例的服务器端中网页呈现设备的结构图;

[0030] 图3示出了根据本发明一个实施例的浏览器中网页呈现设备的结构图;

[0031] 图4示出了根据本发明一个实施例的服务器端中网页呈现方法的流程图;

[0032] 图5示出了根据本发明一个实施例的浏览器中网页呈现方法的流程图;

[0033] 图6示出了现有的一种在线调查问卷的呈现界面;以及

[0034] 图7示出了根据本发明实施例的技术方案实现的一种在线调查问卷的呈现界面。

具体实施方式

[0035] 下面将参照附图更详细地描述本公开的示例性实施例。虽然附图中显示了本公开的示例性实施例,然而应当理解,可以以各种形式实现本公开而不应被这里阐述的实施例所限制。相反,提供这些实施例是为了能够更透彻地理解本公开,并且能够将本公开的范围完整的传达给本领域的技术人员。

[0036] 图1示出了根据本发明一个实施例的网页呈现系统的结构图。如图1所示,网页呈现系统包括多个客户端110(图中示出了3个)、为这些客户端110提供各种网络资源的服务器端120,客户端110和服务器端120经由互连网络而相互通信。

[0037] 在本发明实施例中,客户端110可以是本领域任何可以呈现从服务器端120获取的网络资源的设备,包括但不限于桌面型计算机、笔记式计算机、个人数字助理、智能移

动终端和平板电脑等。客户端 110 上运行有浏览器,所述浏览器能够呈现从服务器端 120 获取的网页文件(例如 html 文件)。

[0038] 服务器端 120 可以根据数据存储器 130 中存储的数据内容来构造网页文件和脚本文件,其中,数据存储器 130 例如为数据库中存储的数据内容包括多个数据项目。服务器端 120 所构造的网页文件包括多个第一页面元素和多个第二页面元素(例如为按钮),每个第一页面元素包括数据内容中的一个数据项目,每个第二页面元素包括对脚本文件中脚本的引用,所引用的脚本适于根据各数据项目之间的逻辑关系切换当前显示的页面元素。

[0039] 需要说明的是,数据存储器 130 可以部署在与服务器端 120 通信连接的设备中,也可以直接部署在服务器端 120,本发明实施例对此不做限制。另外,服务器端 120 所构造的脚本文件可以是独立的文件,也可以是网页文件的一部分(即网页文件中包括各种脚本),本发明实施例对此也不做限制。

[0040] 用户通过客户端 110 中的浏览器访问服务器端 120,服务器端 120 应浏览器的请求而将所请求的网页文件和脚本文件返回给浏览器,为便于网页的显示,服务器端 120 通常还会将与网页相关的其他文件例如 CSS(Cascading Style Sheets,层叠样式表)文件一并返回给浏览器。

[0041] 在本发明实施例中,服务器端 120 构造的网页文件为静态网页文件,该静态网页文件中包括需要在浏览器中进行分页呈现的多个数据项目。也就是说,服务器端 120 是一次性将需要在浏览器中分页呈现的所有数据内容都返回给浏览器。

[0042] 浏览器获取到网页文件和脚本文件后,解析所述网页文件,对所述多个第一页面元素中的一个页面元素进行呈现,并在所呈现的第一页面元素之上呈现相应的第二页面元素。如前所述,所述网页文件中包括有需要在浏览器中进行分页呈现的多个数据项目,根据对网页文件的设计,浏览器解析所述文件后,根据解析结果,会呈现多个数据项目中的一个初始数据项目,即将所述初始数据项目在当前网页中进行呈现。

[0043] 用户浏览完当前呈现的数据项目(有可能还对该数据项目进行相关的操作),为使得在下一个网页中呈现下一个数据项目,会触发当前在第一页面元素中呈现的第二页面元素,例如,单击第一页面元素中的按钮。浏览器检测到对第二页面元素的触发时,执行第二页面元素引用的脚本,以隐藏当前显示的第一页面元素和第二页面元素,并呈现与当前隐藏的第一页面元素相关联的第一页面元素和对应的第二页面元素。如此,实现了对多个数据项目的分页显示。

[0044] 根据本发明实施例的网页呈现系统,通过一次生成静态网页文件(例如 html 文件),数据内容在浏览器端一次下载完成,并通过在浏览器端执行脚本来根据数据内容中各数据项目之间的逻辑关系实现数据内容的分页呈现,由于不需要与服务器端进行多重交互,因而提高了浏览器的网页显示速度,并降低了服务器端的负载。

[0045] 以下对本发明实施例的网页呈现系统中的服务器端 120 以及客户端 110 中浏览器的具体构成进行详细说明,其中,所述服务器端 120 中包括网页呈现设备 200,所述浏览器中包括网页呈现设备 300。

[0046] 图 2 示出了根据本发明一个实施例的服务器端 120 中网页呈现设备 200 的结构图。如图 2 所示,网页呈现设备 200 包括构造单元 210 和响应单元 220。

[0047] 构造单元 210 适于根据包括多个数据项目的数据内容构造网页文件和脚本文件,

所述数据内容可以来自于前述的数据存储器 130。其中，所构造的网页文件包括多个第一页面元素和多个第二页面元素，每个第一页面元素包括数据内容中的一个数据项目，每个第二页面元素包括对脚本文件中脚本的引用，所引用的脚本（例如 JS 脚本）适于根据各数据项目之间的逻辑关系切换当前显示的页面元素。

[0048] 所述第一页面元素和第二页面元素可以采用 <div> 标签，这样，所述脚本文件中的脚本可以通过更改 <div> 标签的”display”属性来切换当前显示的页面元素。具体地，通过将 <div> 标签的”display”属性设置为”display:none”来隐藏该 <div> 标签，通过将 <div> 标签的”display”属性设置为”display:block”来呈现该 <div> 标签。

[0049] 所述数据内容可以包括第一数据条目列表和第二数据条目列表，且每个数据项目包括第一数据条目列表中的一个数据条目和第二数据条目列表中与 该第一数据条目对应的第二数据条目。

[0050] 例如，当所述数据内容为调查问卷的试题时，第一数据条目列表可以为调查问卷的试题的题目表，第二数据条目列表可以为调查问卷的试题的问题选项表。这样，每个数据项目就包括题目表中的一个题目和问题选项表中与该题目对应的选项内容。

[0051] 当所述数据内容包括第一数据条目列表和第二数据条目列表的情况下，所述第一页面元素对应的 <div> 标签可以为嵌套结构，其中外层 <div> 标签的内容为数据内容所在的页码，内层 <div> 标签的内容为第一数据条目以及与该第一数据条目对应的第二数据条目，所述第二页面元素对应的 <div> 标签的内容为按钮，所述按钮引用的脚本适于根据对第一数据条目对应的第二数据条目的选择结果切换当前显示的页面元素。所述按钮可以包括“继续”按钮和“返回”按钮，当用户点击“继续”按钮时，在网页中显示当前页面的下一个页面，当用户点击“返回”按钮时，在网页中显示当前页面上一个页面。

[0052] 继续以所述数据内容为调查问卷的试题为例，构造单元 210 所构造的网页文件中的 <div> 标签示例如下：

[0053]

```
<div id="page_1" style="display:block;">
  <div id="topic1" style>...</div>
</div>
<div id="page_center_1" style="display:block;" class="textcenter mt30">...</div>
<div id="page_2" style="display:none;">
  <div id="topic2" style>...</div>
</div>
<div id="page_center_2" style="display:none;" class="textcenter mt30">...</div>
.....
<div id="page_6" style="display:none;">
  <div id="topic6" style>...</div>
</div>
<div id="page_center_6" style="display:none;" class="textcenter mt30">...</div>
<div id="divsubmit" style="display:none;">...</div>
```

[0054] 在上述代码中，<div> 标签的命名格式如下：

[0055] 页码 <div> 标签格式 :page_+” 页码索引 (以 1 开始)”

[0056] 问卷试题 <div> 标签格式 :topic+” 试题 id”

[0057] 继续、返回按钮 :page_center_+” 页码索引 (以 1 开始)”

[0058] <div> 标签的嵌套逻辑为 :同一页试题包含在页码 <div> 标签中,继续、返回按钮与页码 <div> 标签同级。

[0059] 另外,在上述代码中,“divsubmit” 标签的用途是提交按钮的 div。

[0060] 可以看出,构造单元 210 所构造的网页文件中包括 6 个数据项目,第 1 个数据项目对应的 <div> 标签的 “display” 属性为 “display:block”,为初始要显示的数据项目,其他 5 个数据项目对应的 <div> 标签的 “display” 属性为 “display:none”,为初始在网页中所隐藏的数据项目。

[0061] 如前所述,浏览器中的脚本适于根据各数据项目之间的逻辑关系切换当前显示的页面元素。其中,所述逻辑关系可以是各数据项目之间的自然顺序,也可以是各数据项目之间的逻辑顺序。延续上述调查问卷的例子,数据内容包括 5 个数据项目,其自然顺序为数据项目 1、数据项目 2、数据项目 3、数据项目 4、数据项目 5。而逻辑顺序可能包括 :数据项目 1、数据项目 2、数据项目 3,以及,数据项目 1、数据项目 4、数据项目 5。例如,数据项目 1 的第一数据条目为题目表中的题目 1,与该第一数据条目对应的第二数据条目为题目 1 的选项内容,该选项内容提供两个选项,分别为选项 1 和选项 2,当用户进行在线答题时,对题目 1 的答案为选项 1 时,根据上述逻辑顺序,在下一页中会呈现数据项目 2,对题目 1 的答案为选项 2 时,根据上述逻辑顺序,在下一页中会呈现数据项目 4。

[0062] 响应单元 220 适于根据浏览器的请求将所构造的网页文件和脚本文件返回给浏览器。

[0063] 根据本发明实施例的网页呈现设备 200,通过一次生成静态网页文件 (例如 html 文件),数据内容在浏览器端一次下载完成,并通过在浏览器端执行脚本来根据数据内容中各数据项目之间的逻辑关系实现数据内容的分页呈现,由于不需要与服务器端进行多重交互,因而提高了浏览器的网页显示速度,并降低了服务器端的负载。

[0064] 图 3 示出了根据本发明一个实施例的浏览器中网页呈现设备 300 的结构图。如图 3 所示,网页呈现设备 300 包括请求单元 310、解析单元 320 和处理单元 330。

[0065] 请求单元 310 适于从服务器端获取网页文件和脚本文件。所述网页文件和脚本文件即上述实施例中由服务器端根据包括多个数据项目的数据内容构造得到的网页文件和脚本文件。

[0066] 解析单元 320 适于解析所述网页文件,对所述多个第一页面元素中的一个页面元素进行呈现,并在所呈现的第一页面元素之上呈现相应的第二页面元素。

[0067] 处理单元 330 适于当检测到对第二页面元素的触发时,执行第二页面元素引用的脚本,以隐藏当前显示的第一页面元素和第二页面元素,并呈现与当前隐藏的第一页面元素相关联的第一页面元素和对应的第二页面元素。

[0068] 用户通过客户端 110 中的浏览器访问服务器端 120,服务器端 120 中的响应单元 220 应请求单元 310 的请求而将所请求的网页文件和脚本文件返回给浏览器,为便于网页的显示,服务器端 120 通常还会将与网页相关的其他文件例如 CSS (Cascading Style Sheets,层叠样式表) 文件一并返回给浏览器。

[0069] 在本发明实施例中,服务器端 120 构造的网页文件为静态网页文件,该静态网页文件中包括需要在浏览器中进行分页呈现的多个数据项目。也就是说,服务器端 120 是一次性将需要在浏览器中分页呈现的所有数据内容都返回给浏览器。

[0070] 浏览器获取到网页文件和脚本文件后,解析单元 320 解析所述网页文件,对所述多个第一页面元素中的一个页面元素进行呈现,并在所呈现的第一页面元素之上呈现相应的第二页面元素。如前所述,所述网页文件中包括有需要在浏览器中进行分页呈现的多个数据项目,根据对网页文件的设计,浏览器解析所述文件后,根据解析结果,会呈现多个数据项目中的一个初始数据项目,即将所述初始数据项目在当前网页中进行呈现。

[0071] 用户浏览完当前呈现的数据项目(有可能还对该数据项目进行相关的操作),为使得在下一个网页中呈现下一个数据项目,会触发当前在第一页面元素中呈现的第二页面元素,例如,单击第一页面元素中的按钮。处理单元 330 检测到对第二页面元素的触发时,执行第二页面元素引用的脚本,以隐藏当前显示的第一页面元素和第二页面元素,并呈现与当前隐藏的第一页面元素相关联的第一页面元素和对应的第二页面元素。如此,实现了对多个数据项目的分页显示。

[0072] 根据本发明实施例的网页呈现设备 300,通过一次生成静态网页文件(例如 html 文件),数据内容在浏览器端一次下载完成,并通过在浏览器端执行脚本来根据数据内容中各数据项目之间的逻辑关系实现数据内容的分页呈现,由于不需要与服务器端进行多重交互,因而提高了浏览器的网页显示速度,并降低了服务器端的负载。

[0073] 以下分别介绍服务器端和浏览器中的网页呈现设备所执行的方法。

[0074] 图 4 示出了根据本发明一个实施例的服务器端中网页呈现方法的流程图。如图 4 所示,所述网页呈现方法始于步骤 S410。在步骤 S410 中,根据包括多个数据项目的数据内容构造网页文件和脚本文件,其中所构造的网页文件包括多个第一页面元素和多个第二页面元素,每个第一页面元素包括数据内容中的一个数据项目,每个第二页面元素包括对脚本文件中脚本的引用,所引用的脚本适于根据各数据项目之间的逻辑关系切换当前显示的页面元素。

[0075] 根据一种实施方式,所述第一页面元素和第二页面元素为 <div> 标签,且所述脚本通过更改 <div> 标签的"display"属性来切换当前显示的页面元素。可选地,所述数据内容包括第一数据条目列表和第二数据条目列表,且每个数据项目包括第一数据条目列表中的一个数据条目和第二数据条目列表中与该第一数据条目对应的第二数据条目。

[0076] 可选地,所述第一页面元素对应的 <div> 标签为嵌套结构,其中外层 <div> 标签的内容为数据内容所在的页码,内层 <div> 标签的内容为第一数据条目以及与该第一数据条目对应的第二数据条目,所述第二页面元素对应的 <div> 标签的内容为按钮,所述按钮引用的脚本适于根据对第一数据条目对应的第二数据条目的选择结果切换当前显示的页面元素。

[0077] 需要说明的是,步骤 S410 中所执行的处理与服务器端中的构造单元 210 所执行的处理相同,具体可参照上文对构造单元 210 的描述,这里不做赘述。

[0078] 随后,在步骤 S420 中,根据浏览器的请求将所构造的网页文件和脚本文件返回给浏览器,以供浏览器解析所述网页文件,对所述多个第一页面元素中的一个页面元素进行呈现,并在所呈现的第一页面元素之上呈现相应的第二页面元素,以及,当检测到对第二页

面元素的触发时,执行第二页面元素引用的脚本,以隐藏当前显示的第一页面元素和第二页面元素,并呈现与当前隐藏的第一页面元素相关联的第一页面元素和对应的第二页面元素。

[0079] 需要说明的是,步骤 S420 中所执行的处理与服务器端中的响应单元 220 所执行的处理相同,具体可参照上文对响应单元 220 的描述,这里不做赘述。

[0080] 图 5 示出了根据本发明一个实施例的浏览器中网页呈现方法的流程图。如图 5 所示,所述网页呈现方法始于步骤 S510。在步骤 S510 中,从服务器端获取网页文件和脚本文件,所述网页文件和脚本文件由服务器端根据包括多个数据项目的数据内容构造得到,其中所构造的网页文件包括多个第一页面元素和多个第二页面元素,每个第一页面元素包括数据内容中的一个数据项目,每个第二页面元素包括对脚本文件中脚本的引用,所引用的脚本适于根据各数据项目之间的逻辑关系切换当前显示的页面元素。

[0081] 根据本发明的一种实施方式,所述第一页面元素和第二页面元素为 <div> 标签,且所述脚本通过更改 <div> 标签的" display" 属性来切换当前显示的页面元素。

[0082] 可选地,所述数据内容包括第一数据条目列表和第二数据条目列表,且每个数据项目包括第一数据条目列表中的一个数据条目和第二数据条目列表中与该第一数据条目对应的第二数据条目。

[0083] 可选地,所述第一页面元素对应的 <div> 标签为嵌套结构,其中外层 <div> 标签的内容为数据内容所在的页码,内层 <div> 标签的内容为第一数据条目以及与该第一数据条目对应的第二数据条目,所述第二页面元素对应的 <div> 标签的内容为按钮,所述按钮引用的脚本适于根据对第一数据条目对应的第二数据条目的选择结果切换当前显示的页面元素。

[0084] 需要说明的是,步骤 S510 中所执行的处理与浏览器中的请求单元 310 所执行的处理相同,具体可参照上文对请求单元 310 的描述,这里不做赘述。

[0085] 随后,在步骤 S520 中,解析所述网页文件,对所述多个第一页面元素中的一个页面元素进行呈现,并在所呈现的第一页面元素之上呈现相应的第二页面元素

[0086] 需要说明的是,步骤 S520 中所执行的处理与浏览器中的解析单元 320 所执行的处理相同,具体可参照上文对解析单元 320 的描述,这里不做赘述。

[0087] 在步骤 S520 中呈现了第一和第二页面元素之后,方法进入步骤 S530。在步骤 S530 中,当检测到对第二页面元素的触发时,执行第二页面元素引用的脚本,以隐藏当前显示的第一页面元素和第二页面元素,并呈现与当前隐藏的第一页面元素相关联的第一页面元素和对应的第二页面元素。

[0088] 需要说明的是,步骤 S530 中所执行的处理与浏览器中的处理单元 330 所执行的处理相同,具体可参照上文对处理单元 330 的描述,这里不做赘述。

[0089] 以下给出本发明的一个应用实例,该应用实例为一个在线调查系统。本系统使用 JQuery 公共类库,采用 Microsoft Visual Studio2010 作为开发平台,生成 html 文件,发布到 web 服务器,以供浏览器进行访问使用。

[0090] 首先,以数据库方式来设计数据内容,数据库中包括两张表,分别为题目表和问题选项表。

[0091] 题目表的主要结构为:

- [0092] QuestionId:问卷 Id, Title:题目标题, RelatedId: 关联 Id, PageIndex:页码。
- [0093] 具体内容例如为下表所示。
- [0094]

	QuestionId	Title	RelatedId	PageIndex
1	281	您家里是否有仅供自己或家庭使用的私家车?	1	1
2	281	最近是否有购车计划	3	2
3	281	首次计划购买的私家车价位	4	3
4	281	您是否有购车或者再购车的打算?	5	4
5	281	您再次购车的排量是?	6	5

- [0095] 问题选项表的主要结构为：
- [0096] QuestionTopicOption:选项内容, QuestionRelatedid 选项所关联的题的 Id。
- [0097] 例如“您家里是否有仅供自己或家庭使用的私家车”选无时, QuestionRelatedid 关联题为 3,也就是选项为“无”时,显示 Relatedid 为 3 的题“最近是否有购车计划”。
- [0098] 具体内容例如为下表所示。
- [0099]

	QuestionId	QuestionTopicId	QuestionTopicOption	QuestionRelatedid
1	281	1	无	0.3
2	281	1	有	0.5
3	281	2	1辆	0
4	281	2	2辆	0
5	281	2	3辆	0
6	281	2	4辆以上	0
7	281	3	暂时没有	0
8	281	3	计划购买	0.4
9	281	4	5万以下	0
10	281	4	5-10万	0
11	281	4	10万以上	0
12	281	5	无	0
13	281	5	有	0.6
14	281	6	1.0L及以下	0
15	281	6	1.1-1.6L	0
16	281	6	1.6L以上	0

- [0100] 然后,就可以根据上述两个表的结构构造网页文件和脚本文件。
- [0101] 其中,生成的 html 文件的部分代码如下：
- [0102]

```
private string GetTopicHtml(DataTable dttopic, string name, string bd, short isUseAuthCode, short
userAgreement, string pvScript, bool progress)
{
    string[] ridArray = strqrId.Split(',');
    DataView dv = null;
    DataTable dt = null;
    string brandFunction = "";
    string dateFunction = "";
    StringBuilder sbHtml = new StringBuilder();
    string onsubmitFunction = string.Empty;
    sbHtml.AppendLine("<body>");
}
```

[0103]

```

sbHtml.AppendLine("<input id=\"qidPV\" name=\"qidPV\" type=\"hidden\" value=\"\" + Qid + \"\"
/>");

sbHtml.AppendLine("<div class=\"coheader\"><a href=\"#\"></a></div>");

sbHtml.AppendLine("<form id=\"Question\" name=\"form7\" method=\"post\" target=\"_self\" \"
+ onSubmitFunction + \" action =\"\" + actionUrl + Qid + \"&style=Survey&backurl=\" + newUrl + \"s_\" + Qid +
\".html\" >");

sbHtml.AppendLine("<input type=\"hidden\" id =\"usetime\" name=\"usetime\" />");
sbHtml.AppendLine("<div class=\"comain\"><div class=\"ctit\">");
sbHtml.AppendLine("<h2 class=\"fonyh\">\" + name + \"</h2>");
//问题开始标签
sbHtml.AppendLine("<div class=\"mcont\"><div class=\"c838383\" id=\"questionContent\">\" +
bd + \"</div>");
//sbHtml.AppendLine("<ul class=\"Q_list\">");
int qcount = dttopic.Rows.Count;
int topicIndex = 0; //题号计数器
//页码
int pageIndex = 0;
//添加分页隐藏域标识
sbHtml.AppendLine("<input type=\"hidden\" id=\"queryType\" value=\"fanyeNo\" />");
#region 题目
int rows = 1;
for (int i = 0; i < qcount; i++)
{
    string tid = dttopic.Rows[i][\"Id\"].ToString();
    string tName = CreateHtml.GetHtmlEncode(dttopic.Rows[i][\"Title\"].ToString());
    string qtype = dttopic.Rows[i][\"QuestionTopicType\"].ToString();
    string qrid = dttopic.Rows[i][\"RelatedId\"].ToString();
    int imgColumns =
RequestQueryString.StrToInt(dttopic.Rows[i][\"imgColumns\"].ToString());
    string displ = \"\";
    int pageIndexTemp = 0;
    string pageBeginStr = string.Empty;
    string pageEndStr = string.Empty;
    if (CreateHtml.IsExists(ridArray, qrid))
    {
        displ = \"display:none;\";
    }
    else
    {
        topicIndex++;
    }

    #region 分页html
    //是否分页显示
    if (showModel == 1)
    {
        pageTopicSet += string.Format(\"{0}-\", tid);
        int pi = Convert.ToInt32(dttopic.Rows[i == qcount - 1 ? i : i +
[] [\"PageIndex\"].ToString());
        if (rows == int.Parse(dttopic.Rows[i][\"PageIndex\"].ToString()) && rows != pi)
        {
            pageTopicSet += \", \";
        }
        rows = pi;
        //第一页div
        if (pageIndex == 0)

```

[0104]

```

    {
        pageIndex = int.Parse(dttopic.Rows[i]["PageIndex"].ToString());
        pageBeginStr = string.Format("<div id='page_{0}' >", pageIndex);
        sbHtml.Append(pageBeginStr);
    }
    pageIndexTemp = int.Parse(dttopic.Rows[i]["PageIndex"].ToString());

    //判断分页结束 添加结束</div>
    //并添加“继续”按钮
    if (pageIndex != pageIndexTemp)
    {
        if (pageIndex == 1)
        {
            string styledisplay = string.Empty;

            pageEndStr = string.Format("</div><div id='page_center_' + pageIndex + \"
style='display:none;'\" class='txtcenter mt30'><a class='ydown ml40' \" + styledisplay + \"
id='a_page_{0}_1' onclick='ShowPage({0},1);'>继续</a></div><div id='page_{1}' style='display:none;'>",
                pageIndex, pageIndex + 1);
        }
        else
        {
            pageEndStr = string.Format("</div><div id='page_center_' + pageIndex + \"
style='display:none;'\" class='txtcenter mt30'><a class='ydown ml40' id='a_page_{0}_0'
onclick='ShowPage({0},0);' style='display:none;'>返回</a>&nbsp;&nbsp;&nbsp;<a class='ydown ml40'
id='a_page_{0}_1' onclick='ShowPage({0},1);' style='display:none;'>继续</a></div><div id='page_{1}'
style='display:none;'>",
                pageIndex, pageIndex + 1);
        }
        pageIndex = pageIndexTemp;

        sbHtml.Append(pageEndStr);
        pageEndStr = "";
    }
}

#endregion

#region 添加题目选项
sbHtml.AppendLine("<div id='topic' + qrid + \"\" style='\" + displ + \"\">");
sbHtml.AppendLine("<a name='tp' + qrid + \"\"></a>");
sbHtml.AppendLine("<h3><span class='d60000 df1' Data='\" + qrid + \"\"> + topicIndex
+ ".</span><span class='df1'><em class='em1'>\" + tName + \"</em></span>");
sbHtml.AppendLine("</h3>");
#endregion

#region 自定义题目

//选择题
if (qtype.Equals("1"))
{
    //DisplayLogic 为前端js显示逻辑关联的试题
    //qrid (被关联的试题的rid) rid本试题的rid, id本试题选项的id属性名称
    sbHtml.Append("<input name='QuestionTopic9615' type='radio' id='answer46846'
onclick='DisplayLogic(' 0,3', '1', 'answer46846', 'AutoHomeLogin', '0')' value='46846'>");
}
#endregion

```

[0105]

```

sbHtml.AppendLine("</div>");
#region 分页结束html
    if (i + 1 == qcount)
    {
        ++pageIndexTemp; //最后一题了pageEndStr =
        pageEndStr =
            string.Format("</div><div id=\"page_center_\" + pageIndex + \"
style=\"display:none;\" class=\"txtcenter mt30\"><a class=\"ydown ml40\" id='a_page_{0}_0'
onclick=' ShowPage({0},0);' style='display:none;'>返回</a> <input type=\"submit\" class=\"buttjs\"
value=\"提交问卷\" /><input type=\"hidden\" id=\"maxPage\" value=\"{0}\" >\", pageIndex);
        pageEndStr += "</div>";
    }
    sbHtml.Append(pageEndStr);
    pageEndStr = "";
#endregion
}
#endregion

sbHtml.AppendLine("</div>");
sbHtml.AppendLine("</div>");
sbHtml.AppendLine("</form></div>");
sbHtml.AppendLine("</body></html>");
return sbHtml.ToString();
}
}

```

生成的脚本文件的部分 js 脚本如下:

//显示逻辑关联的试题, 其中, qrid (被关联的试题的关联rid), rid (本试题的关联rid), id (本试题选项的id 属性名称)

```

function DisplayLogic(qrid, rid, id, fname, isLogin) {

    var hdvl = $("#hd" + rid).val();
    var oldr = hdvl;
    var newr = qrid;
    if ($("#" + id).attr("type") == "radio") {

        $("#hd" + rid).val(qrid);
        // isAllowInput(id); //单选的时候处理填空项效果
        //当单选选项关联同一个试题的时候, 这个试题不做任何处理
        newr = GetNewRid(qrid, oldr);
        oldr = GetNewRid(oldr, qrid);
        // eachNum();
    }
    else {
        if ($("#" + id).attr("checked")) {
            $("#hd" + rid).val(hdvl + "," + qrid);
            oldr = "0";
            // eachNum();
        } else {
            $("#hd" + rid).val(hdvl.replace(",", " + qrid, ""));
            oldr = qrid;
            newr = "0";
        }
    }
    oldr = GetUpdateRid(oldr);
    CloseLogic(oldr, newr);
}
}

```

[0106]

```

//当单选选项时, 获取新的试题id
function GetNewRid(rid1, rid2) {
    var rid1Arr = rid1.split(",");
    var rid2Arr = rid2.split(",");
    var newrid = "0";
    for (var a = 0; a < rid1Arr.length; a++) {
        var bol = false;
        for (var b = 0; b < rid2Arr.length; b++) {
            if (rid1Arr[a] == rid2Arr[b]) {
                bol = true;
                // $("#topic" + rid1Arr[b]).attr("style").remove();
                break;
            }
        }
        if (!bol) {
            newrid = newrid + "," + rid1Arr[a];
        }
    }
    return newrid;
}

//关闭显示出来的试题
function CloseLogic(olddrid, newrid) {
    var oldArr = olddrid.split(",");
    var newArr = newrid.split(",");
    for (var i = 0; i < oldArr.length; i++) {
        for (var j = 0; j < arrQuestion.length; j++) {
            if (oldArr[i] == arrQuestion[j].RId.toString()) {
                if (arrQuestion[j].Count > 0) {
                    arrQuestion[j].Count--;
                    break;
                }
            }
        }
    }
    for (var i = 0; i < newArr.length; i++) {
        for (var j = 0; j < arrQuestion.length; j++) {
            if (newArr[i] == arrQuestion[j].RId.toString()) {
                arrQuestion[j].Count++;
                break;
            }
        }
    }
    for (var i = 0; i < arrQuestion.length; i++) {
        if (arrQuestion[i].RId != 0) {
            if (arrQuestion[i].Count == 0) {
                $("#topic" + arrQuestion[i].RId).css("display", "none");
                $("input", $("#topic" + arrQuestion[i].RId)).each(function () {
                    if ($(this).attr("type") == "hidden") {
                        $(this).val("");
                    }
                    if ($(this).attr("type") == "radio" || $(this).attr("type") == "checkbox") {
                        //$(this).attr("checked", false);
                    }
                    if ($(this).attr("type") == "text") {
                        if ($(this).attr("name").indexOf("tk") > -1) {
                            // $(this).val("填空");
                            //$(this).attr("disabled", "disabled");
                        }
                    }
                });
            }
        }
    }
}

```

[0107]


```

        }
        else {
            $(this).val("");
        }
    }
});
$("#select", $("#topic" + arrQuestion[i].RId)).each(function () {
    // $(this).get(0).selectedIndex = 0;
});
$("#textarea", $("#topic" + arrQuestion[i].RId)).each(function () {
    $(this).val("");
});
} else {
    $("#topic" + arrQuestion[i].RId).css("display", "block");
}
}
}
var num = 0;
$(".d60000").each(function () {
    if ($(this).attr("class") == "d60000 df1") {
        if ($(this).parent().parent().parent().css("display") != "none") {
            num++;
            $(this).html(num + ".");
        }
    }
});
}

```

[0108] 上述 js 脚本的执行逻辑为：点击“继续”进入下一页；隐藏当前页内容；页码加 1；判断当前页是否显示；若是，显示当前页，若否，页码加 1 后继续进行上述的判断。

[0109] 之后，就可以根据浏览器的请求将上述 html 文件和脚本文件返回给浏览器，以供浏览器对调查问卷中的试题进行分页显示。

[0110] 在线调查问卷在浏览器中的一种呈现界面如图 7 所示，在该呈现界面中，当用户输入完成身份证号后点击“继续”按钮，则浏览器不需要再向服务器端进行 http 请求，直接呈现下一页的内容，当用户输入完成身份证号后点击“返回”按钮，则浏览器也不需要再向服务器端进行 http 请求，直接呈现上一页的内容。

[0111] 根据上述应用实例，调查问卷试题由浏览器一次下载完成，然后在浏览器前端分页显示，另外，该调查问卷还提供关联试题显示，根据相应选项显示后面的试题。这样，使用发明实施例的技术方案后，在线调查问卷用户可以快速打开，以分页方式答题，改善了用户交互体验，并减少了与服务器的 http 请求的次数。

[0112] 在此提供的算法和显示不与任何特定计算机、虚拟系统或者其它设备固有相关。各种通用系统也可以与基于在此的示教一起使用。根据上面的描述，构造这类系统所要求的结构是显而易见的。此外，本发明也不针对任何特定编程语言。应当明白，可以利用各种编程语言实现在此描述的本发明的内容，并且上面对特定语言所做的描述是为了披露本发明的最佳实施方式。

[0113] 在此处所提供的说明书中，说明了大量具体细节。然而，能够理解，本发明的实施例可以在没有这些具体细节的情况下实践。在一些实例中，并未详细示出公知的方法、结构和技术，以便不模糊对本说明书的理解。

[0114] 类似地,应当理解,为了精简本公开并帮助理解各个发明方面中的一个或多个,在上面对本公开的示例性实施例的描述中,本公开的各个特征有时被一起分组到单个实施例、图、或者对其的描述中。然而,并不应将该公开的方法解释成反映如下意图:即所要求保护的本发明要求比在每个权利要求中所明确记载的特征更多的特征。更确切地说,如下面的权利要求书所反映的那样,发明方面在于少于前面公开的单个实施例的所有特征。因此,遵循具体实施方式的权利要求书由此明确地并入该具体实施方式,其中每个权利要求本身都作为本发明的单独实施例。

[0115] 本领域那些技术人员可以理解,可以对实施例中的设备中的模块进行自适应性地改变并且把它们设置在与该实施例不同的一个或多个设备中。可以把实施例中的模块或单元或组件组合成一个模块或单元或组件,以及此外可以把它分成多个子模块或子单元或子组件。除了这样的特征和/或过程或者单元中的至少一些是相互排斥之外,可以采用任何组合对本说明书(包括伴随的权利要求、摘要和附图)中公开的所有特征以及如此公开的任何方法或者设备的所有过程或单元进行组合。除非另外明确陈述,本说明书(包括伴随的权利要求、摘要和附图)中公开的每个特征可以由提供相同、等同或相似目的的替代特征来代替。

[0116] 此外,本领域的技术人员能够理解,尽管在此所述的一些实施例包括其它实施例中所述的某些特征而不是其它特征,但是不同实施例的特征的组合意味着处于本发明的范围之内并且形成不同的实施例。例如,在下面的权利要求书中,所要求保护的实施例的任意之一都可以以任意的组合方式来使用。

[0117] 本公开的各个部件实施例可以以硬件实现,或者以在一个或者多个处理器上运行的软件模块实现,或者以它们的组合实现。本领域的技术人员应当理解,可以在实践中使用微处理器或者数字信号处理器(DSP)来实现根据本发明实施例的网页呈现设备中的一些或者全部部件的一些或者全部功能。本发明还可以实现为用于执行这里所描述的方法的一部分或者全部的设备或者装置程序(例如,计算机程序和计算机程序产品)。这样的实现本公开的发明程序可以存储在计算机可读介质上,或者可以具有一个或者多个信号的形式。这样的信号可以从因特网网站上下载得到,或者在载体信号上提供,或者以任何其他形式提供。

[0118] 应该注意的是上述实施例对本发明进行说明而不是对本发明进行限制,并且本领域技术人员在不脱离所附权利要求的范围的情况下可设计出替换实施例。在权利要求中,不应将位于括号之间的任何参考符号构造成对权利要求的限制。单词“包含”不排除存在未列在权利要求中的元件或步骤。位于元件之前的单词“一”或“一个”不排除存在多个这样的元件。本发明可以借助于包括有若干不同元件的硬件以及借助于适当编程的计算机来实现。在列举了若干装置的单元权利要求中,这些装置中的若干个可以是通过同一个硬件项来具体体现。单词第一、第二、以及第三等的使用不表示任何顺序。可将这些单词解释为名称。

[0119] 本发明公开了:

[0120] A1、一种网页呈现方法,适于在浏览器中执行,所述方法包括:

[0121] 从服务器端获取网页文件和脚本文件,所述网页文件和脚本文件由服务器端根据包括多个数据项目的数据内容构造得到,其中所构造的网页文件包括多个第一页面元素和多个第二页面元素,每个第一页面元素包括数据内容 中的一个数据项目,每个第二页面元

素包括对脚本文件中脚本的引用,所引用的脚本适于根据各数据项目之间的逻辑关系切换当前显示的页面元素;

[0122] 解析所述网页文件,对所述多个第一页面元素中的一个页面元素进行呈现,并在所呈现的第一页面元素之上呈现相应的第二页面元素;以及

[0123] 当检测到对第二页面元素的触发时,执行第二页面元素引用的脚本,以隐藏当前显示的第一页面元素和第二页面元素,并呈现与当前隐藏的第一页面元素相关联的第一页面元素和对应的第二页面元素。

[0124] A2、如 A1 所述的网页呈现方法,其中,所述第一页面元素和第二页面元素为 <div> 标签,且所述脚本通过更改 <div> 标签的" display" 属性来切换当前显示的页面元素。

[0125] A3、如 A2 所述的网页呈现方法,其中,所述数据内容包括第一数据条目列表和第二数据条目列表,且每个数据项目包括第一数据条目列表中的一个数据条目和第二数据条目列表中与该第一数据条目对应的第二数据条目。

[0126] A4、如 A3 所述的网页呈现方法,其中,所述第一页面元素对应的 <div> 标签为嵌套结构,其中外层 <div> 标签的内容为数据内容所在的页码,内层 <div> 标签的内容为第一数据条目以及与该第一数据条目对应的第二数据条目,所述第二页面元素对应的 <div> 标签的内容为按钮,所述按钮引用的脚本适于根据对第一数据条目对应的第二数据条目的选择结果切换当前显示的页面元素。

[0127] B5、一种网页呈现方法,适于在服务器端执行,所述方法包括:

[0128] 根据包括多个数据项目的数据内容构造网页文件和脚本文件,其中所构造的网页文件包括多个第一页面元素和多个第二页面元素,每个第一页面元素包括数据内容中的一个数据项目,每个第二页面元素包括对脚本文件中脚本的引用,所引用的脚本适于根据各数据项目之间的逻辑关系切换当前显示的页面元素;

[0129] 根据浏览器的请求将所构造的网页文件和脚本文件返回给浏览器,以供浏览器解析所述网页文件,对所述多个第一页面元素中的一个页面元素进行呈现,并在所呈现的第一页面元素之上呈现相应的第二页面元素,以及,当检测到对第二页面元素的触发时,执行第二页面元素引用的脚本,以隐藏当前显示的第一页面元素和第二页面元素,并呈现与当前隐藏的第一页面元素相关联的第一页面元素和对应的第二页面元素。

[0130] B6、如 B5 所述的网页呈现方法,其中,所述第一页面元素和第二页面元素为 <div> 标签,且所述脚本通过更改 <div> 标签的" display" 属性来切换当前显示的页面元素。

[0131] B7、如 B6 所述的网页呈现方法,其中,所述数据内容包括第一数据条目列表和第二数据条目列表,且每个数据项目包括第一数据条目列表中的一个数据条目和第二数据条目列表中与该第一数据条目对应的第二数据条目。

[0132] B8、如 B7 所述的网页呈现方法,其中,所述第一页面元素对应的 <div> 标签为嵌套结构,其中外层 <div> 标签的内容为数据内容所在的页码,内层 <div> 标签的内容为第一数据条目以及与该第一数据条目对应的第二数据条目,所述第二页面元素对应的 <div> 标签的内容为按钮,所述按钮引用的脚本适于根据对第一数据条目对应的第二数据条目的选择结果切换当前显示的页面元素。

[0133] C9、一种网页呈现设备,驻留在浏览器中,所述设备包括:

[0134] 请求单元,适于从服务器端获取网页文件和脚本文件,所述网页文件和脚本文件

由服务器端根据包括多个数据项目的数据内容构造得到,其中所构造的网页文件包括多个第一页面元素和多个第二页面元素,每个第一页面元素包括数据内容中的一个数据项目,每个第二页面元素包括对脚本文件中脚本的引用,所引用的脚本适于根据各数据项目之间的逻辑关系切换当前显示的页面元素;

[0135] 解析单元,适于解析所述网页文件,对所述多个第一页面元素中的一个页面元素进行呈现,并在所呈现的第一页面元素之上呈现相应的第二页面元素;以及

[0136] 处理单元,适于当检测到对第二页面元素的触发时,执行第二页面元素引用的脚本,以隐藏当前显示的第一页面元素和第二页面元素,并呈现与当前隐藏的第一页面元素相关联的第一页面元素和对应的第二页面元素。

[0137] C10、如 C9 所述的网页呈现设备,其中,所述第一页面元素和第二页面元素为 <div> 标签,且所述脚本通过更改 <div> 标签的" display" 属性来切换当前显示的页面元素。

[0138] C11、如 C10 所述的网页呈现设备,其中,所述数据内容包括第一数据条目列表和第二数据条目列表,且每个数据项目包括第一数据条目列表中的一个数据条目和第二数据条目列表中与该第一数据条目对应的第二数据条目。

[0139] C12、如 C11 所述的网页呈现设备,其中,所述第一页面元素对应的 <div> 标签为嵌套结构,其中外层 <div> 标签的内容为数据内容所在的页码,内层 <div> 标签的内容为第一数据条目以及与该第一数据条目对应的第二数据条目,所述第二页面元素对应的 <div> 标签的内容为按钮,所述按钮引用的脚本适于根据对第一数据条目对应的第二数据条目的选择结果切换当前显示的页面元素。

[0140] D13、一种网页呈现设备,驻留在服务器端,所述设备包括:

[0141] 构造单元,适于根据包括多个数据项目的数据内容构造网页文件和脚本文件,其中所构造的网页文件包括多个第一页面元素和多个第二页面元素,每个第一页面元素包括数据内容中的一个数据项目,每个第二页面元素包括对脚本文件中脚本的引用,所引用的脚本适于根据各数据项目之间的逻辑关系切换当前显示的页面元素;

[0142] 响应单元,适于根据浏览器的请求将所构造的网页文件和脚本文件返回给浏览器,以供浏览器解析所述网页文件,对所述多个第一页面元素中的一个页面元素进行呈现,并在所呈现的第一页面元素之上呈现相应的第二页面元素,以及,当检测到对第二页面元素的触发时,执行第二页面元素引用的脚本,以隐藏当前显示的第一页面元素和第二页面元素,并呈现与当前隐藏的第一页面元素相关联的第一页面元素和对应的第二页面元素。

[0143] D14、如权 D13 所述的网页呈现设备,其中,所述第一页面元素和第二页面元素为 <div> 标签,且所述脚本通过更改 <div> 标签的" display" 属性来切换当前显示的页面元素。

[0144] D15、如 D14 所述的网页呈现设备,其中,所述数据内容包括第一数据条目列表和第二数据条目列表,且每个数据项目包括第一数据条目列表中的一个数据条目和第二数据条目列表中与该第一数据条目对应的第二数据条目。

[0145] D16、如 D15 所述的网页呈现设备,其中,所述第一页面元素对应的 <div> 标签为嵌套结构,其中外层 <div> 标签的内容为数据内容所在的页码,内层 <div> 标签的内容为第一数据条目以及与该第一数据条目对应的第二数据条目,所述第二页面元素对应的 <div> 标

签的内容为按钮,所述按钮引用的脚本适于根据对第一数据条目对应的第二数据条目的选择结果切换当前显示的页面元素。

[0146] E17、一种网页呈现系统,所述系统包括浏览器和服务端,所述浏览器与所述服务端通信连接,且所述浏览器包括如 C9-12 中任一项所述的网页呈现设备,所述服务端包括如 D13-16 中任一项所述的网页呈现设备。

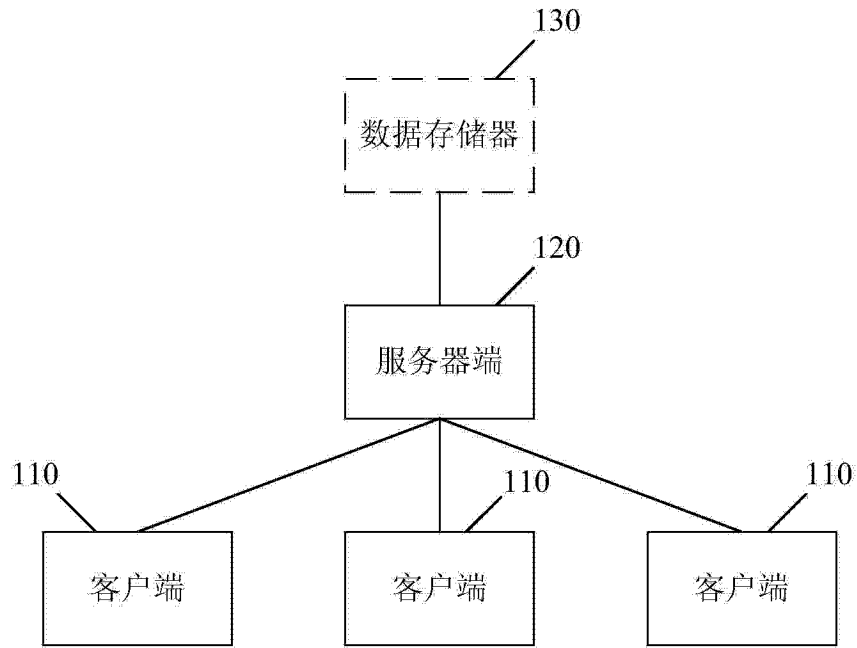


图 1

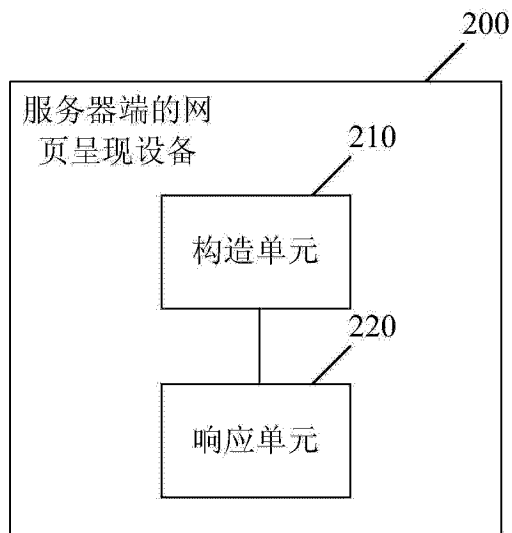


图 2

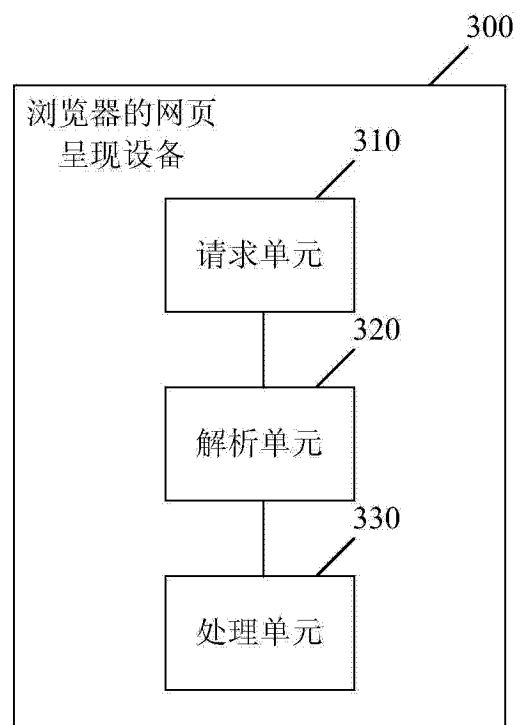


图 3

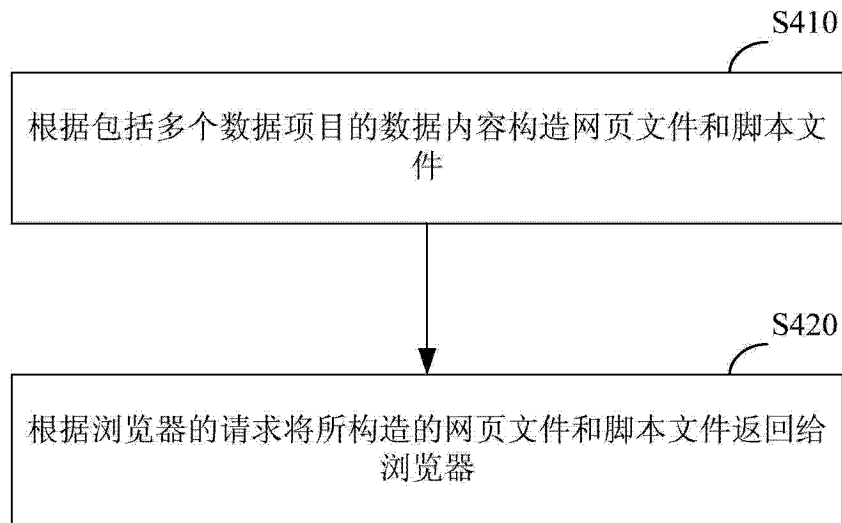


图 4

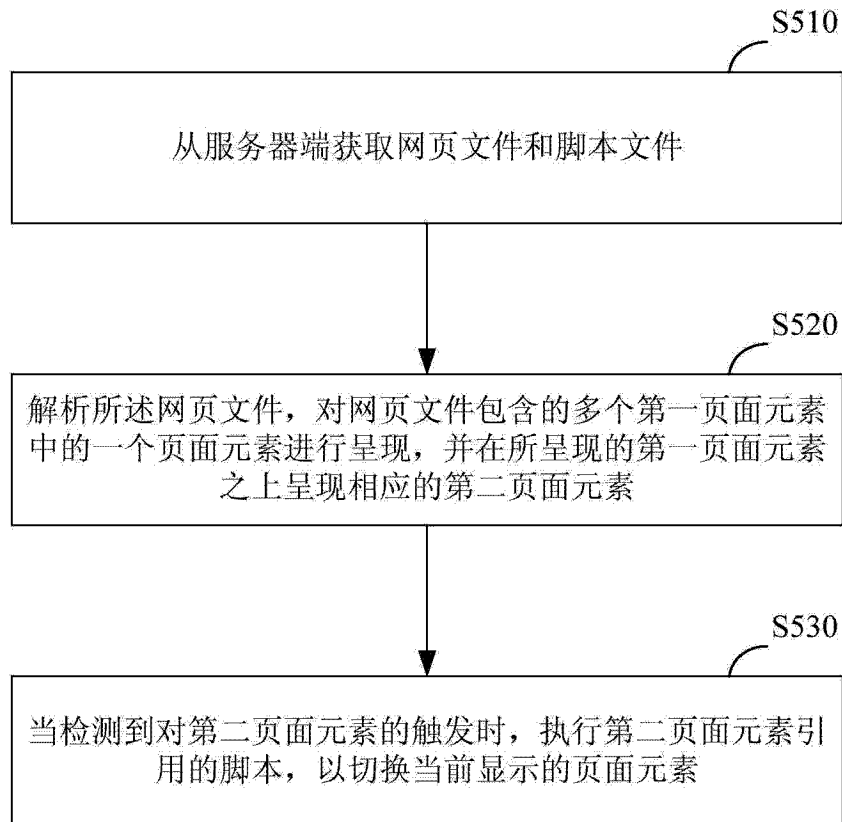


图 5

第一部分 员工基本信息

1. 您的性别为？※（单选题）*

- A: 男 B: 女

2. 您的受教育程度是？※（单选题）*

- A: 大专以下 B: 大专 C: 本科 D: 硕士 E: 博士

3. 您的年龄是？※（单选题）*

- A: 25岁（含）以下 B: 25岁~35岁（含） C: 35岁~45岁（含） D: 45岁~55岁（含） E: 55岁以上

4. 您在本公司的工作年限为？※（单选题）*

- A: 1年（含）以下 B: 1年~3年（含） C: 3年~5年（含） D: 5年~10年（含） E: 10年以上

图 6

汽车之家用户访问习惯调查问卷

2. 身份证号 *

返回

继续

图 7