

信息发布终端网络通信方法的研究

Research of Network Communication Technique on Message Posting Terminal

吴文娟 鄢 萍 孟祥涛 (重庆大学 机械工程学院制造工程研究所 重庆 400030)

摘要: 针对单一的推送技术不能实现和用户的互动以及单一的拉取技术及时性差的问题,本文在深入分析 Linux 网络通信的基础上,提出了一种使信息发布终端能够采用推送和拉取相结合的方法与服务器通信的终端通信软件的架构方法,在该方法中设计了一种基于套接口的信息发布终端与服务器通信的异步通信协议,并提出了一些保持通信稳定性的措施。

关键词: 信息发布终端 网络通信 套接口 通信协议

1 引言

信息发布终端的主要功能是发布视频、音频、文字、图片等广告信息,它通常运用在人流密集区域,如车站、机场、商场、超市等公共场合。当前信息发布终端与服务器的通信方式主要有两种。一种是推送的方式,即按照指定的时间间隔或根据发生的事件把选定的内容自动推送给消费者^[1]。这是一种早已存在的信息获取方式,如传统的无线电广播与电视就采用这种方式,信息推送技术实际上是传统的广播式媒体与网络结合的产物^[2]。这种推送的方式是有一定弊端的,它脱离了消费者的需要,依靠盲目的海量投放各种广告来唤起消费者的注意,往往转瞬即逝,很难满足消费者进一步获取信息的需求。另一种是拉动的方式,是指消费者有目的地在网上主动查询信息,通过终端的浏览器获取所需信息^[3]。这种方式虽然具有很强的针对性,能满足用户个性化的需求,但其及时性差,对用户的要求较高,并且信息发布终端在消费者不需要查询信息时处于空闲状态,其使用率低。

本课题所研发的信息发布终端为了解决上述问题,提出了一种推送和拉取相结合的通信方式。其在通常状况下使用推送的方式,由服务器控制,将信息发送给各个终端结点,当消费者需要查询信息时,采用拉取的方式,由信息发布终端直接向服务器发起请求,满足不同消费者的特殊需要。

2 信息发布系统的整体结构

整个信息发布系统由信息发布终端、局域网、

服务器组成,信息发布终端分布在各个广告点,服务器负责节目分发和对终端的管理。信息发布终端与服务器采用 C/S 结构,通过网络进行数据传输,一个局域网范围内仅需要一台服务器^[4]。服务器端采用 windows 操作系统,使得维护人员操作更简便;信息发布终端采用嵌入式 Linux 操作系统,低功耗,而可靠性高。

终端信息发布系统应具有以下几个功能模块(图 1) 终端自身的管理、与服务器通信及下载要发布的信息、视频播放、图片/文本信息显示、信息查询。各模块相互独立,并行运行,模块间采用管道、信号及文件共享的方式进行通信。

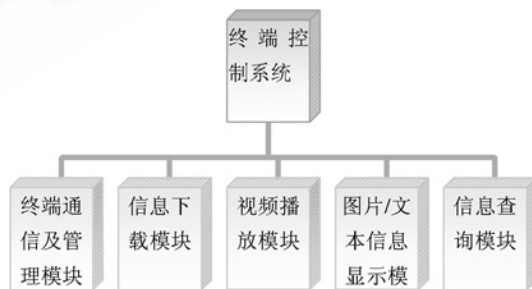


图 1 软件系统结构图

终端通信及管理模块是核心的部分,服务器采用推送的方式与其通信,信息发布终端接收服务器发送的消息,解析并执行其命令(图 2);

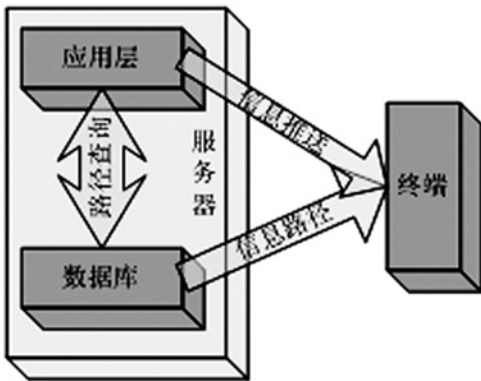


图 2 信息推送

而当消费者进行查询操作时,信息查询模块将查询的关键字传递给终端通信及管理模块,此时服务器采用拉取的方式与信息发布终端通信,根据查询关键字将相关信息返回给信息发布终端(图3)。同时,终端通信及管理模块还要负责对自身系统的管理,对其它几个进程的运行状态进行监控,使它们按照计划的动作执行,并处理在与服务器通信过程中产生的各种突发异常状况。

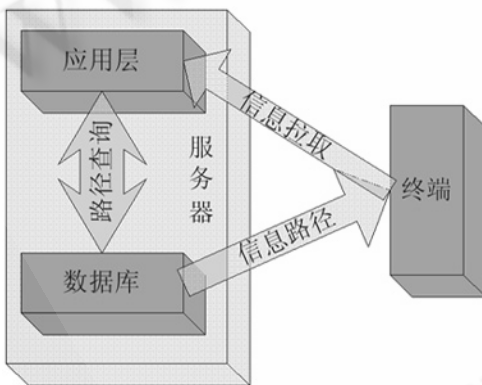


图 3 信息拉取

3 信息发布终端与服务器间的应用层通信协议

此协议规定了信息发布终端与服务器在不同通信方式下的通信规则。此协议应满足表达清晰、易于解析、通信可靠、工作效率高等要求,同时也要考虑其灵活性和可扩展性。根据信息发送的主体,协议包括服务器和信息发布终端两大部分,根据通信方式不同,协议又可分为信息推动、信息拉取和终端控制三大部分。

协议规定:每条信息由包头、包尾、和描述符3块内容组成,包头为类别标志符,表示协议的种类,包尾

为/加类别标志符,表示该条信息的结束。类别描述符包括:

定时同步——CONTROL;

推送信息——PUSH;

拉取信息——PULL;

描述符表示此条协议所包含的含义,允许有一个或者多个。

基本格式:

| | | | | |
|-------------------------------------|-------------------|-------------------|----------------|----------|
| <类别标志符> (CONTROL/ PUSH/ PULL) | 描 述 符 1; | 描 述 符 2; | | </类别标志符> |
|-------------------------------------|-------------------|-------------------|----------------|----------|

3.1 定时同步方式下的协议

1) 时间校准信息。因为在信息发布终端需要准确显示时间,并且需要定时开关机和定时播放预定节目,所以终端系统时间必须和服务器时间严格保持一致性。采用的办法就是终端定时请求服务器校准时钟,服务器收到请求后首先获取自身系统时钟再将此时钟传给终端。

终端协议:

| | | |
|----------------------|-----------------|------------------------|
| <类别标志符> (CONTROL) | 时钟描述符 (TIME) | </类别标志符> (/CONTROL) |
|----------------------|-----------------|------------------------|

服务器端协议:

| | | | |
|----------------------|-----------------|----------|------------------------|
| <类别标志符> (CONTROL) | 时钟描述符 (TIME) | 时钟 内容 | </类别标志符> (/CONTROL) |
|----------------------|-----------------|----------|------------------------|

2) 心跳包。为了便于终端判断服务器的连接状态,终端要定时向服务器发送心跳包以期望得到服务器的响应。

终端协议:

| | | |
|----------------------|-----------------------|------------------------|
| <类别标志符> (CONTROL) | 心跳包描述符 (HEARTBEAT) | </类别标志符> (/CONTROL) |
|----------------------|-----------------------|------------------------|

服务器端协议:

| | | |
|----------------------|------------------|------------------------|
| <类别标志符> (CONTROL) | 服务器状态描述符 (OK) | </类别标志符> (/CONTROL) |
|----------------------|------------------|------------------------|

3) 信息分类表。为了使消费者在终端查询到的信息分类表始终与服务器保持一致,终端需要定时请求服务器发送最新信息分类表。此部分协议的描述符

应包括信息分类表的描述符和存储路径。

终端协议：

| | | |
|------------------------|----------------------|--------------------------|
| <类别标志符> (CONTROL) | 信息分类表描述符 (LIST) | </类别标志符> (/CONTROL) |
|------------------------|----------------------|--------------------------|

服务器端协议：

| | | | |
|------------------------|----------------------|------|--------------------------|
| <类别标志符> (CONTROL) | 信息分类表描述符 (LIST) | 存储路径 | </类别标志符> (/CONTROL) |
|------------------------|----------------------|------|--------------------------|

3.2 信息推送方式下的协议

文件更新信息：当信息发布终端没有进入查询状态时，服务器以推送的方式将需要更新的文件名和存储路径发送给终端，终端收到后自动到服务器上下载此文件，下载完成后向服务器报告。为了保证下载文件的正确性和完整性，终端在下载完成后计算出文件的 md5 校验码，然后与从服务器发送来的校验码比较，如果不同则重新下载此文件。此部分协议的描述符应包括文件类型描述符(表 1)、存储路径及 md5 校验码。

表 1 文件类型描述符表

| | 音/视频 | 图片 | 系统配置文件 | 文本信息 |
|--------|------|-----|--------|------|
| 文件类型标记 | PRG | IMG | CNF | TXT |

例如，音/视频文件的传输协议如下所示：

| | | | | |
|---------------------|--------------------|-------------------------------|---------|-----------------------|
| <类别标志符> (PUSH) | 文件类型描述符 (PRG) | 存储路径 (/home/prg /1. mpg) | md5 校验码 | </类别标志符> (/PUSH) |
|---------------------|--------------------|-------------------------------|---------|-----------------------|

3.3 信息拉取方式下的协议

查询关键字和查询信息：当信息发布终端进入查询状态后，服务器以拉取的方式和终端进行通信。信息发布终端将查询的关键字发送给服务器，服务器收到后返回所查询信息的存储路径，终端按照此路径到服务器上下载需要信息，并在下在完成向服务器报告。

此部分协议的终端类描述符包括所查询信息的类别描述符和查询关键字，此部分协议的服务器类描述符包括查询信息在服务器中的存储路径及信息文件的 md5 校验码。

信息拉取方式下的终端传输协议如下：

| | | | | |
|---------------------|---------|---------|---------|-----------------------|
| <类别标志符> (PULL) | 查询关键字 1 | 查询关键字 2 | 查询关键字 3 | </类别标志符> (/PULL) |
|---------------------|---------|---------|---------|-----------------------|

信息拉取方式下的服务器端传输协议如下

| | | | | |
|---------------------|---------|------|---------|-----------------------|
| <类别标志符> (PULL) | 最后一级关键字 | 存储路径 | md5 校验码 | </类别标志符> (/PULL) |
|---------------------|---------|------|---------|-----------------------|

4 信息发布终端与服务器间的通信实现及通信可靠性考虑

4.1 通信实现

4.1.1 信息发布终端拉取技术的实现

在服务器端，对终端用户需要查询的信息数据分类进行存储。因为数据量较大，为了减少查询时间，存储时可分为多级树形存储目录，最低一级目录中存放供查询信息，如型号、生产厂家、价格、性能参数等。然后按树的遍历的方法形成信息分类表，存放指定路径。同时，在服务器索引数据库中，也同样按照此树形目录分类，存储各数据项在服务器中的存储路径。

信息发布终端与服务器连通以后，在终端设定一个专用定时器，每次时间片结束时到服务器指定路径下载更新前面形成的信息分类表。最后，信息发布终端根据最新下载的信息分类表来定制个性化的适用于不同环境的检索界面，以供用户查询。

信息发布终端检索界面的每级目录均为一个查询关键字，当消费者进行查询操作时，信息发布终端以拉取的方式与服务器通信，将消费者查询的最后一级目录以及前面的各级目录都作为关键字发送给服务器，为了提高服务器的查询效率，所有关键字按级数从高到低进行存储。如 3.3 节电子产品查询系统，如果用户查询的最后一级目录为 AT89C51 单片机，则发送给服务器的关键字包括：元器件——单片机——ATMEL——AT89C51，终端协议为：

| | | | | | |
|---------------------|-------|-------|-------|---------|-----------------------|
| <类别标志符> (PULL) | 元 器 件 | 单 片 机 | ATMEL | AT89C51 | </类别标志符> (/PULL) |
|---------------------|-------|-------|-------|---------|-----------------------|

服务器接收到后，在分类存储信息的数据库中逐级查找与关键字匹配的条目，数据库中包括所有数据信息在服务器中的存储路径。如上例，首先在最高一

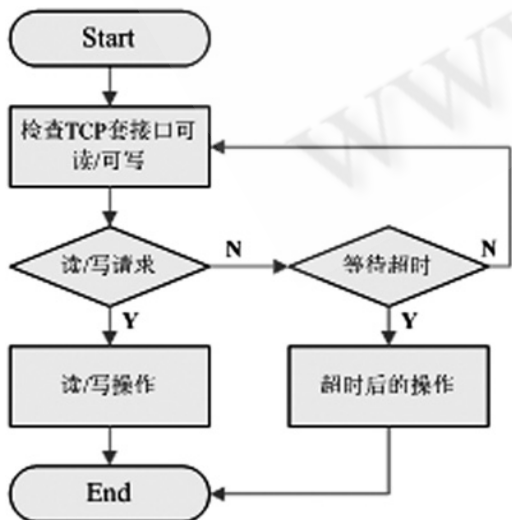
级数据库中查找元器件,在元器件类中查找其子类单片机类,直到查找到最后一级 AT89C51 为止,最后得到该信息的存储路径。服务器将查找到的内容发送给终端,然后终端就将信息提供给消费者,以达到信息拉取的目的。上例中,服务器端协议为:

| | | | | |
|-------------------|----------------------|------|------------|---------------------|
| <类别标志符> (PULL) | 最后一级关键字 (AT89C51) | 存储路径 | md5 校验码 | </类别标志符> (/PULL) |
|-------------------|----------------------|------|------------|---------------------|

4.1.2 如何实现推拉技术的结合

为了实现与服务器通信,信息发布终端首先需要建立一条到服务器的连接,此连接可以采用完全阻塞或非阻塞两种方式。如果采用完全阻塞的方式,则系统会阻塞在对 TCP 套接口某种操作(读或写)上,如果是阻塞于从服务器获得推送过来的信息,则终端不能把需要服务器拉取的信息及时的发送给服务器;反之在写时阻塞也一样;更不幸的时,如果当终端阻塞在读/写操作时,刚好网络不通,或服务器已经停止响应,则除非重起,终端就永远停止在这个操作上了。而如果采用非阻塞得方式,则信息发布终端系统就会过于频繁的读写到服务器的通信接口,给终端系统造成很大的负担。

因此,为了避免上述情况的发生,实现推送与拉取相结合,程序中采用了 I/O 复用技术(图 4)。采用该技术以后,系统不会阻塞于对 TCP 套接口的读或写操作上,当服务器有信息推动到终端或终端有信息需要发送给服务器时,终端系统都能够即刻进行处理,因而实现了信息推送与拉取技术的结合。



4.2 通信可靠性考虑

为了实现控制终端信息的推送和拉取,信息发布终端与服务器必须保持可靠的连接。因此信息发布终端必须对服务器的状态进行实时监控,及时对各种突发故障(如网络断开、服务器死机、服务器重起、终端死机等)进行处理。这主要是靠终端给服务器发送信息从而主动探测服务器的网络状态来实现。在系统运行过程中,终端会不断请求服务器向其发送心跳包,如果终端在指定时间内没有收到服务器的心跳包,就需要探测服务器的运行状态。这时终端向服务器主动发送探测信息,观测服务器,在不同的状态下服务器会有不同的反应^[7],见表 2:

表 2 终端网络连接故障表

| | 错误信息 | 返回值 |
|----------|----------------------|-----|
| 网络不可达 | "No route to host" | <0 |
| 服务器崩溃 | "No route to host" | <0 |
| 服务器重起 | "Connection refused" | =0 |
| 服务器关闭该进程 | "Connection refused" | =0 |
| 正常情况 | 无 | >0 |

为了区分不同的异常情况,只需对不同的返回值进行处理就行了,如下:

当服务器重起,或者服务器已经关闭对终端的响应进程,即返回值 =0 时,终端需要关闭当前建立的连接,重新建立到服务器连接;

当网络不可达,或服务器崩溃,即返回值 <0 时,需要继续监测此套接口,直到网络接通或者服务器重新运行为止;

4.3 应用案例

本信息发布终端已应用于电子产品市场广告查询终端中。该终端机具有音/视频广告播放,图片、文本类广告显示,以及信息查询等功能。当服务器采用推送技术与终端机通信时,终端机滚动播放服务器推送的所有信息,以达到广告的效果;当终端用户需要查询信息时,服务器采用拉取的方式与终端通信,终端迅速从服务器下载用户所需信息,达到信息查询的效果(如图 5 为电子产品信息查询系统的效果图,每个查询项及其子项均为查询关键字)。

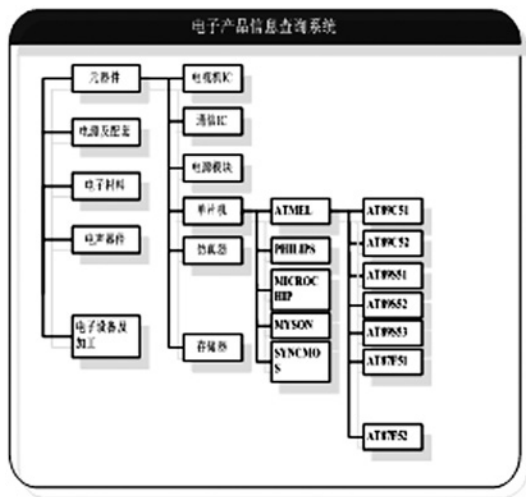


图 5 (电子产品市场)信息查询系统效果图

5 结束语

不同于传统的信息发布系统单一推送和单一拉取的方式,本文提出了一种在 Linux 系统上采用推送和拉取相结合的方式与服务器在局域网通信的机制,在兼顾到软件开发工作量和可扩展性的情况下,制定了终端和服务器在应用层的通信协议。还采取了 I/O 复用技术以实现信息推送与拉取的结合,使终端不会阻塞于单纯的推送或则拉取操作上。最后,为了保证终

端与服务器的通信可靠性,考虑到可能发生的各种异常情况,均给出了相应的处理办法,从而实现终端的智能化。

参考文献

- 1 刘迎清. 国内信息推送研究综述. 长沙大学学报, 2006, 20(5): 82-86.
- 2 李秀. 推送技术——完善网络信息服务的关键技术. 现代图书情报技术, 2001, (5): 56-59.
- 3 王辉、陈凌、张丽娟. 信息推拉技术. 情报科学, 2004, 22(12): 1440-1443.
- 4 张鸿雁、陈晓春、王元. LINUX 系统下 PC 机局域网并行计算环境的实现与测试. 西安建筑科技大学学报, 39(2): 267-271.
- 5 谢国、王荣良. 基于多媒体广告机安全性的 SSL 研究与实现. 微型电脑应用, 2007, 22(9): 37-39.
- 6 李海文、任静、张李秋. 交通监控系统中信息发布平台的设计与实现. 计算机技术与发展, 2007, 17(4): 52-56.
- 7 Kay A. Robbins, Steven Robbins. Practical UNIX Programming. Prentice Hall/Pearson, 1999.
- 8 Warren W. Gay. 实战 Linux Socket 编程. 西安: 西安电子科技大学出版社, 2002.