

利用 SMS 作为 Internet 应用终端的方法研究

李彦

(重庆工学院网络信息中心 重庆 400050)

摘要 Internet 应用研究中的一个重要问题是如何扩展终端设备,本文研究了利用 SMS 将移动电话扩展为 Internet 应用终端的方式,主要解决了两大问题:在系统架构上引入了 Web Service 作为分布式服务的技术框架;在系统实现上有效解决了发送 SMS 的串行设备与并发任务的异步问题。

关键词 SMS, Web Service, 分布式, Internet

On the Approach of Utilizing SMS as Internet Application System Terminal

LI Yan

(Chongqing Technology Institute, Chongqing 400050)

Abstract The emphasis in internet application research is how to extend terminal. The essay tells about how to use SMS to extend mobile phone to Internet terminal. The essay mainly solves two problems; one is to introduce Web Service to distributed service technology architecture; the other is to resolve the problem of the asynchronism between serial device sending SMS and parallel task.

Keywords SMS, Web Service, Distributed, Internet

1 引言

随着移动电话的普及, SMS (Short Messaging Service) 作为一种无线通信技术已经得到了广泛的应用, 它的最大优势在于无论何时何地均可以十分方便地利用移动电话发送和接收文本信息。在 Internet 应用系统中, 希望将用户终端扩展到所有的手持设备, 而不仅仅局限于桌面计算机, 使得用户在使用网络应用时不受时间地点的限制。移动电话是普及得最广的一种手持终端设备, 某些移动电话具有直接接入 Internet 的功能, 但这种功能由于费用高昂、网络速度慢、显示面积小等缺点而并未随着移动电话的普及而普及。由于 SMS 服务是移动电话服务中使用最普及的一种服务, 可以使用 SMS 服务作为网络应用系统的输出/输入终端。

2 系统运行模式

Internet 应用系统的基本特点是各子系统在功能上分布、在位置上分散, 子系统之间通过网络协议进行相互协作, 因此在设计 SMS 服务时应该考虑到为各种“功能”和“位置”上相异的子系统提供服务。从面向对象的观点来看, 这个 SMS 服务应该是独立于任何一个子系统的、自主提供服务的“对象”, 通过对这个“对象”属性和方法的调用, 达到获取 SMS 服务的目的。在传统的应用中, 提供服务的“对象”和服务使用者在同一台机器的内存中, 这种操作就变得相当简便了; 但在 Internet 应用中, 提供服务的“对象”和服务使用者不仅可能不在同一台机器上, 而且可能不在同一个子网中, 要使用这样的服务“对象”, 必须借助于分布式计算技术。

从上世纪 90 年代开始, 经过一轮又一轮分布式计算技术的竞争, Microsoft 的 DCOM、SUN/Java 的 RMI, 甚至包括 OMG 的 CORBA 都未真正成为事实上的标准, 近几年兴起的 Web Service 技术凭 Web 和 XML 而逐渐成为 Internet 应用系统中分布式计算的新标准。

系统的运行模式可设计为: 利用 Web Service 技术, 通过分布式计算的模式, 向应用系统中各种子系统提供简单方便的发送 SMS 服务; 当收到 SMS 消息后, 也通过主控界面即时

调用应用系统提供的 Web Service, 达到对 SMS 消息进行即时处理的目的。

3 系统架构设计

架构设计主要考虑到以下因素: ①由于 SMS 发送设备是串行设备, 不可能支持多线程, 因此控制 SMS 发送设备的控件对象不能够直接由 Web Service 生成; ②对于 Web Service 多线程同时接收到的并发任务, 转发给 SMS 发送设备时需要转化成串行任务; ③管理员需要一个良好的 SMS 工作监控界面。因此, 在架构设计中除了 Web Service 外, 还引了一个 Web Form 层, 以解决以上三个问题。系统架构如图 1 所示。

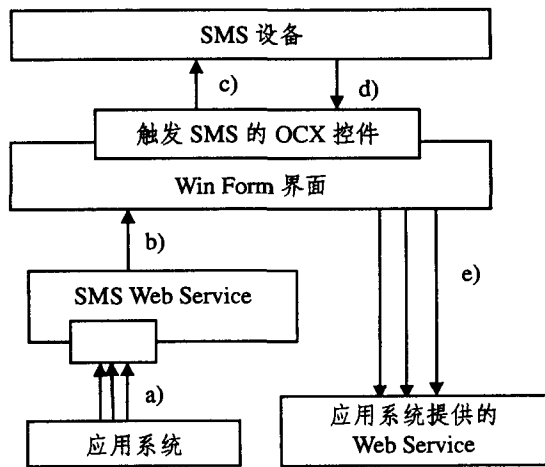


图 1

从架构图上可以看出, 应用系统通过 Web Service 调用接口 a 处进行调用, Web Service 通过接口 b 将需要发送的 SMS 信息传递到 Win Form, 再通过接口 c 将 SMS 信息文本串送到 SMS 设备发出。当 SMS 设备收到一条短消息后, 通过 d 立即触发 OCX 事件, 将收到的 SMS 信息进行解析处理, 并通过接口 e 直接调用相应应用系统的 Web Service, 以达到

对应用系统进行处理的目的,这样就实现了收到短消息后,即时对应用系统进行响应的目的。

4 系统实现

从架构图中来看,系统的实现并无太大困难,但有几点值得注意:①从应用系统的角度来看,发送和接收短消息是异步的,即:应用系统发送一个 SMS 之后,并不能立即得到这个 SMS 的反馈,而且什么时候能够得到反馈是无法控制的;② SMS Web Service 是一个操作系统后台运行的多线程任务,如何即时向 Win Form 传递发送 SMS 的任务? ③由于发送一个短消息的最少用时为 5 秒左右,因此 Win Form 界面中发送消息也不能做到即时发送,即:应用系统请求一个发送任务后, SMS Web Service 能够即时接收这个任务,但 Win Form 并不能保证即时发送这个 SMS,因此在 Win Form 上需要一个“缓冲池”实现机制,将不能即时发送的消息保存下来;④接收到一个短消息后,系统如何知道此条短消息对应于曾经发送的哪条短消息? 需要调用哪个应用系统的 Web Service 对收到的此条短消息进行处理?

为了有效解决以上问题,首先得构建一个 SMS 服务类,记录一条 SMS 的细节,其主要属性如下(本文中所有代码均采用 C# 语言格书写):

```
public class SMS
{
    public long ID;           //自动增长的编号
    public string MobileNo;   //发送手机号
    public string Content;    //短消息内容
    public bool IsSend;       //是否已经发送
    public bool IsOK;         //发送是否成功
    public DateTime BackTime; //收到回复短信时间
    public string BackMobileNo; //回复短信的手机号
    public string BackContent; //回复内容
    public int FunctionType;   //应用系统模块编号,本条短信由哪个应用系统模块进行处理的
    public bool IsHandleSuccess; //相应功能模块是否已成功处理此条短消息
}
```

①考虑到发送和接收短消息的异步性,因此在发送短消息时必须将 SMS 对象保存下来,在接收到该条短消息的回复时再重建此对象,方便系统继续处理。要保存一个活动对象到外存上有许多方法(如:串化技术),这里直接采用数据库进行存放,将一个 SMS 对象保存一条记录,对 ID 字段进行识别和重建。

②使用了数据库技术后, SMS Web Service 向 Win Form 传递 SMS 对象就可以简化为传递一个 SMS 对象的 ID 即可。 SMS Web Service 是后台服务,与 Win Form 最佳方式还是通过获取 Win Form 的 Handle 值后,通过 API 函数 PostMessage() 进行。最简单获取 Win Form 的 Handle 值的方法是在这个 Win Form 运行时将其自身的 Handle 值保存到指定目录下的文本文件中(为了便于存取,最好采用 XML 文件),在 SMS Web Service 中再从 XML 文件中读入 Handle 的值。

在 SMS Web Service 中 PostMessage 函数定义如下:

```
[DllImport("User32.dll", EntryPoint="PostMessage")]
public static extern int PostMessage (int hWnd, uint Msg, int wParam, int lParam);
将 SMS 对象的 ID 值通过 wParam 参数传回到 Win Form 中,自定义消息编号为 0x0501;
PostMessage(handle, 0x0501, Int32.Parse(no), 0);
```

在 Win Form 中增加对编号为 0x0501 的消息进行处理的程序:

```
protected override void DefWndProc (ref System.Windows.Forms.Message m)
{
    switch(m.Msg)
    {
        case 0x0501:
            执行处理消息的程序;
            break;
        default:
            base.DefWndProc(ref m); //调用基类函数处理非自定义消息。
            break;
    }
}
```

③SMS 设备是串行设备,因此 Win Form 收到 SMS 发送任务后,应先送到“缓冲池”排队,再逐一通过 OCX 控件发送到 SMS 设备。“缓冲池”的实现并不难,使用一个列表(List)对象即可实现。需要重点考虑的是从“缓冲池”中读取记录,再通过 OCX 发送这个动作的实现方式。因为此 Win Form 有三个任务需要同时完成:a)接收发送任务;b)通过 OCX 向 SMS 设备发送 SMS 细节;c)接收 SMS 并调用相应应用系统的 Web Service 进行处理。其中 a)、c)任务的响应是由 Windows 系统进程控制,而 b)任务必须由用户自己在 Win Form 上通过程序完成,因此对于“OCX 读取‘缓冲池’并向 SMS 设备发送”这个过程也必须使用线程来实现:

```
private void AddToList(string no)
{
    this.listSendCache.Items.Add(no);
    //每一次向缓冲池中加入记录,判断 SMS 设备当前是否在工作。如果不在工作,则激活线程
    if(! this.isBusy)
    {
        isBusy = true;
        this.tSMS = new Thread(new ThreadStart(SendThread));
        this.tSMS.Start(); //激活线程
    }
}

private void SendThread() //发送 SMS 的线程
{
    if(this.listSendCache.Items.Count == 0)
    {
        isBusy = false;
        return;
    }
    this.listSendCache.SetItemChecked(0, true);
    this.listSendCache.SelectedIndex = 0;
    this.GetSMSDetail(this.listSendCache.Items[0].ToString());
    this.listSendCache.Items.RemoveAt(0);
    SendThread();
    return;
}
```

④发送和接收 SMS 是异步的,对于接收到的 SMS 要解决的问题是如何知道此条回复短消息对应的是曾经发出的哪条 SMS。解决此问题有一个最简单的办法:在发送 SMS 时,附上 SMS 对象的 ID 号,用户回复短消息时,约定先输入此 ID 号,再输入回复的内容。系统收到短消息后,通过解析消息的 ID 号,再调出数据库中的记录重建此 SMS 对象,通过 SMS 对象的 FunctionType 属性来决定该调用哪个应用系统的 Web Service 来处理接收到的短消息。

小结 “网络无处不在”已成为现实,但离“Internet 无处不在”还有些距离,其关键问题在于上网终端的受限。通过 SMS 技术,将手机变成 Internet 应用终端是一种成本低、有实效的方法。由于 SMS 本身的局限性,对于篇幅较长的交流文本并不适合于此,多媒体交流就更难于实现;但据笔者研究,大多数 Internet 应用系统只需要少量的文本即可实现系统交互功能,因此研究如何将 SMS 与 Internet 应用相结合具有极其重要的实用价值。

参考文献

- 马潮. 嵌入式 GSM 短信接口的软硬件设计. 单片机与嵌入式系统应用, 2003(7): 21~24
- 朱燕. 移动网短消息业务平台的建设探讨. 通信世界, 2004(14): 42~43
- Microsoft. Creating and Accessing XML Web Services Walkthroughs. <http://msdn.microsoft.com/library>
- IBM. developerWorks: SOA and Web services. <http://www-106.ibm.com/developerworks/webservices/>
- IBM. 关于 Web 服务应用的个案研究. <http://www-900.cn.ibm.com/developerWorks/cn/webservices/ws-sms/>
- Microsoft. Web 服务策略附件. <http://www.microsoft.com/china/MSDN/library/WebServices/WebServices/Usdnlglobspecws.mspx>
- Guthery S B, Cronin M J. Mobile Application Development with SMS and the SIM Toolkit. USA: McGraw-Hill, 2003
- 周学冰, 石丹. 短消息(SMS)与 WAP 的开发及应用. 北京: 电子工业出版社, 2002