

Smart Client 技术实现 MIS 系统中的离线应用

彭玉卓, 杨开英, 马俊

(武汉理工大学 计算机学院, 湖北 武汉 430070)

摘要: 为满足传统 MIS 系统日益复杂的业务需要, 采用智能客户端的架构。智能客户端是易于部署和管理的客户端应用程序, 它们通过统筹使用本地资源和到分布式数据资源的智能连接, 从而为现实应用提供适应的、快速响应的和丰富的交互式体验, 能轻松实现 MIS 系统中的离线应用。

关键词: smart client; 管理信息系统; offline application block

中图分类号: TP311.13

文献标识码: A

文章编号: 1673-629X(2007)03-0200-03

The Offline Application of MIS with Smart Client

PENG Yu-zhuo, YANG Kai-ying, MA Jun

(Computer School, Wuhan Univ. of Tech., Wuhan 430070, China)

Abstract: In order to satisfy the complicated requirements of MIS system, smart client is adopted to solve these problems. A smart client is an application that uses local processing, consumes XML Web services and can be deployed and updated from a centralized server. The NET framework (Windows forms) provides the ability to develop smart clients with ease. Smart client is the concept of constructing your application solution into a smart, flexible and convenient platform that utilizes Web services for communication. It's easy to make an offline application of MIS.

Key words: smart client; management information system; offline application block

0 引言

目前, 我国的 MIS 已相当普及, 其应用范围几乎覆盖了各个行业和部门。MIS 系统也由最初的单机 C/S 架构转变为业界普遍使用的 B/S 架构, 该架构满足了分布式应用的需要, 较好地满足了企业的需求, 然而随着企业信息化的不断发展、业务的不断增长以及新业务的不断出现, 管理信息系统越来越复杂化。传统的、简单的 MIS 显然已经无法满足管理的需要。有了 Smart Client 技术, 可以很好地将胖客户端和瘦客户端应用的优缺点结合在一起, 来满足实际情况的需要^[1]。Smart Client 可以自动灵活地进行升级和更新, 从而简化了系统的维护; Smart Client 还可以充分地利用本地的计算资源, 将 MIS 处理的载荷合理地分配给系统中的每一台计算机, 从而提高系统的响应和性能。

1 Smart Client 技术介绍

智能客户端^[2]: 智能客户端是易于部署和管理的

客户端应用程序, 它们通过统筹使用本地资源和到分布式数据资源的智能连接, 从而为您提供适应的、快速响应的和丰富的交互式体验。

智能客户端应用程序将下列要素组合在一起, 从而为它的最终用户提供快速响应的、丰富的和吸引人的体验。

1.1 本地资源和用户体验

随着计算机速度的提高, 客户端的计算机能力也有了大大提升, 传统的瘦客户端程序无法充分利用客户端的计算能力和其它资源。而智能客户端应用程序具有的一个显著特性就是具有利用本地资源的能力, 它的代码和数据部署在客户端上并且在本地执行和访问。例如, 访问闪存, 利用扫描仪和摄像头等。智能客户端解决方案通过充分利用 Microsoft Windows 平台提供的所有功能, 为最终用户提供良好的客户端体验。无论您是在线还是离线, 都可以使用智能客户端应用程序。当在线时, 智能客户端应用程序可以提供更加丰富的体验。在离线时, 智能客户端也可以运行得特别好。

1.2 连接

在企业或 Internet 范围内, 智能客户端应用程序能够轻松地与系统连接并与之进行数据交换。Web

收稿日期: 2006-06-13

作者简介: 彭玉卓(1982-), 男, 湖北钟祥人, 硕士研究生, 研究方向为数据库应用技术; 杨开英, 教授, 硕士生导师, 研究方向为数据仓库、数据库。

服务使智能客户端解决方案能够利用行业标准协议(例如,XML,HTTP和SOAP)来与任何类型的远程系统交换信息。在许多情况下,智能客户端应用程序可以向用户提供类似于门户的功能,从而将完全不同的数据和服务加以协调并集成到总体解决方案中。较之于以前的任何技术,Visual Studio,.NET Framework和.NET Compact Framework让Web服务的使用更为容易。Microsoft SQL Server,Microsoft Message Queuing (MSMQ)和BizTalk Server等技术还提供了在系统之间同步和交换信息的易于使用的方式。

1.3 具备离线的能力

在现实环境中,应用程序对网络的依赖性越来越强,但是很多情况下都不可能保证无时无刻存在网络连接。对于智能客户端应用程序而言,无论是否连接到Internet,它都可以正常工作。在没有网络连接或网络连接断续时,智能客户端能够利用本地缓存和处理进行操作。该功能对于新一代的移动信息工作者来说极具价值,尤其是在指定了移动连接的成本、期限和速度的情况下。

离线功能不仅可以在移动方案中使用,而且对于桌面解决方案也可以利用离线体系结构来更新后台线程上的后端系统,从而保持用户界面的响应并改善整体的最终用户体验。该体系结构还能够提供成本和性能上的优势,这是因为用户界面不需要在智能客户端与服务器之间来回切换。

由于智能客户端只能在后台与其他系统交换所需的数据,因此可以让人感到它与其他系统交换的数据量的减少(即使在硬连接的客户端系统上,这种带宽的减少也会带来巨大的好处)。这反过来会增强用户界面(UI)的响应,因为UI不是通过远程系统呈现的。

1.4 智能部署和更新

传统客户端应用程序很难进行部署和更新,许多胖客户端应用程序具有大量复杂的安装要求,并且可能通过注册组件或在公共位置安装DLL来共享代码,从而导致应用程序脆弱性和更新困难。经常会出现这样的情况,安装一个应用程序会中断另一个应用程序。智能客户端使得安装和维护客户端应用程序变得容易。

智能客户端应用程序可以在其运行时或位于后台时对自身进行自动更新,并且.NET框架使应用程序可以相互隔离,以便在安装一个应用程序时不会破坏另一个应用程序,同一应用程序的多个版本可以并列部署。这些功能大大简化了应用程序部署,消除了许多与胖客户端应用程序相关联的应用程序脆弱性问题^[3]。

2 Smart Client 在管理信息系统的离线应用

文中将利用Offline Application Block来实现该离线应用。Offline Application Block是根据.NET框架的功能并封装智能客户端应用程序来构建的,以帮助用户在脱机时执行任务,就像用户在联机时执行任务一样简单而有效。Offline Application Block可以提供具有脱机功能的智能客户端应用程序所需的基本功能。

Offline Application Block主要提供以下服务:通过判断网络适配器的物理连接状态检测网络是否连通,并自动在联机和离线两种模式下切换;缓存不易改变的数据;在机器重新联机的时候执行存储在机器中的请求。以上三个特征使得Offline Application Block具备基本的支持离线应用程序开发的功能。

2.1 问题分析

在MIS系统中,并不是所有的应用都能够保证随时在线,需要保证在明确脱机,使用低带宽或高延迟网络,或者连接时断时续的情况下继续高效地工作。因此,客户端系统对于离线的需求与日俱增。

为了解决上述问题,可以通过Offline Application Block技术实现离线应用,可以构建新的、更强大的应用程序。

2.2 构建可离线应用的MIS系统

由于有了Smart Client技术,可以很好地将胖客户端和瘦客户端应用的优点结合在一起,适应使用实际情况的需要。Smart Client可以自动灵活地进行升级和更新从而简化了系统的维护;Smart Client可以充分利用本地的计算资源,可以将MIS处理的载荷合理地分配给系统中的每一台计算机,提高了系统的响应和性能。

微软公司提供的Microsoft Application Block为开发具有智能更新功能的.NET应用提供了极大的便利。在MIS系统中重用并扩展了Offline Application Block(OAB)^[4]等应用程序模块,实现了符合MIS应用实际需求的自动更新等功能。

使用OAB可以实现对.NET应用智能更新支持,OAB为应用提供了下载、验证和后置处理机制。通过OAB提供的接口,可以轻易对OAB根据自己需要进行扩展。在MIS系统中,使用BITS下载机制,保证系统的运作效率。OAB的工作流程如图1所示。

OAB主要由四个子系统组成:连接状态管理、服务代理管理、参考数据管理、消息数据管理。

(1)连接状态管理。此部分消息注册机制可以使程序调用者得到当前的网络状态是处于联机状态还是脱机状态。一旦网络状态发生改变的时候,可以及时得到响应,并依此来调整相应的业务逻辑。有两种方

法可以判断连接状态:手动判断或通过自动过程判断。

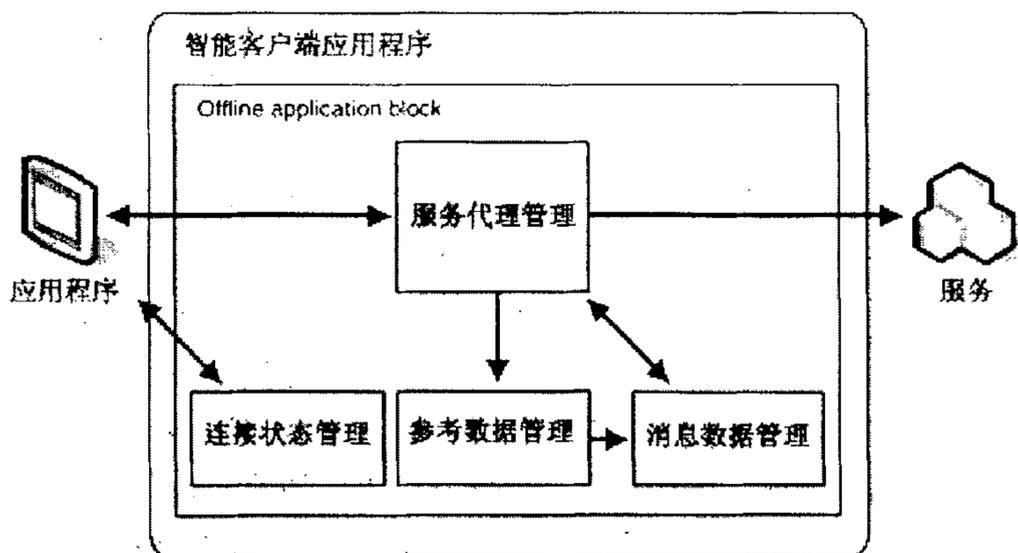


图 1 OAB 工作流程

(2)服务代理管理。此模块管理两类数据:一类是消息数据管理;另一类是参考数据管理。同时它还负责同服务器进行交互。它会进行协调,以便将任务完成通知返回到应用程序。

(3)参考数据管理。它与服务代理管理和消息数据管理配合工作,以下载存储在本地计算机上的参考数据。在大多数情况下,参考数据是用于完成工作流的只读数据。参考数据管理可使参考数据与服务器上的数据保持一致。它将消息存储在队列中以下载参考数据。然后,执行程序将使用消息服务请求与服务连接;以下载参考数据。

(4)消息数据。它是在工作流过程中创建的数据。当应用程序处于脱机状态时,该数据将存储在一个本地队列中。当应用程序联机后,执行程序会从队列中删除消息,发出与服务器同步数据的服务请求,然后数据就会与服务器进行同步。

要在应用程序中使用 Offline Application Block,首先添加一个对 Common, Offline Providers 和 Offline 程序集的引用。

```

.....
using Microsoft. ApplicationBlocks. SmartClient. Offline;
.....

```

通过调用 OfflineBlockBuilder. Instance 属性从 OfflineBlockBuilder 类创建一个块生成器实例。使用 OfflineBlockBuilder 类构建并访问该应用程序块的所有组件。

```

.....
private OfflineBlockBuilder offlineBlockBuilderInstance;
offlineBlockBuilderInstance = OfflineBlockBuilder. Instance;
.....

```

使用“连接管理器”组件注册联机/脱机更改状态事件,如下面的代码所示:

```

.....
offlineBlockBuilderInstance. ConnectionManager. ConnectionState

```

```

ChangedEvent += new ConnectionStateChangedEventHandler(Con-
nectionManager ConnectionStateChangedEvent);
.....

```

为脱机客户端创建一个“应用程序服务代理”的实例。使用“应用程序服务代理”为各种事件注册控制器。这些事件可通知最终用户在后台发生的更改。如果原始的“应用程序服务代理”不可用,则使用 FailsafeServiceAgent 注册控制器以便处理错误。

```

.....
FailsafeServiceAgent refFailSafeServiceAgent =
offlineBlockBuilderInstance. FailsafeServiceAgent;
refFailSafeServiceAgent. ErrorEvent += new Meth-
odExecutionFailureReportEventHandler ( FailSafeServiceAgentError
Event);
.....

```

调用 OfflineBlockBuilder 对象的 Start 方法来启动“连接管理器”。该组件将轮询连接状态,并在连接状态发生更改时将事件发送到“服务代理”。

```

.....
offlineBlockBuilderInstance. Start();
.....

```

如果要强制应用程序在脱机模式下工作,请使用 ConnectionManager 组件的 GoOffline 方法,如下面的代码所示:

```

.....
offlineBlockBuilderInstance. ConnectionManager. GoOffline();
.....

```

3 结束语

Smart Client 智能客户端可以动态升级、自动更新,可以方便地经 Web 运行而不用担心防火墙问题并可以方便地实现 MIS 系统的离线运用,方便地连接 WebServices^[5],大大降低了应用实现的技术风险,节省了项目开发周期。

参考文献:

[1] 明俊峰. Smart Client 在管理信息系统中的应用[J]. 电脑知识与技术,2005(3):77 - 80.
[2] Microsoft Corporation. Smart Client Architecture and Design Guide[M]. USA:Microsoft Press,2004.

(下转第 206 页)

交叉路口 Junction 的集合。Route 是由五元组(标识号 id, 路径长 l , 折线段 c , kind, start) 表示, kind 表示路径的单双行信息, start 中有两种情况(smaller, larger), 表示在二维平面中 xy 顺序下, 路径起始点相对比另一端点的前后。交叉路口是一个依赖于路径的三元组 $Junction(R) = (rm1, rm2, cc)$, $rm1$ 和 $rm2$ 表示连接于该交叉点的路径及交叉点位于路径的位置, 该位置由路径上的长度表示。 cc 是整数类型的转弯矩阵, 标注道路的交通规则, 如 U 型转弯、左转弯、右转弯等。

设置 $Side = \{up, down, none\}$, 其中 up, down 表示双行线的两个方向, none 表示是一条单行线。交叉路口由两条相交路径表示, 若有两条以上的路径交于一点, 可用每两条路径表示一次这一点。即当交叉于同一交叉路口的路径多于 2 条时, 这一交叉路口将要被重复表示。且 cc 对于单行线没有实际意义, 当相交于同一点的路径都是单行线时可忽略。

3 结论及展望

虽然人们对 GIS 的数据模型和数据结构进行了大量的研究, 开发了许多商业化软件, 但目前 GIS 软件没有统一完备的数据模型, 即使 GIS 数据模型的概念也没有统一的认识。而空间数据模型是空间数据库的基础和核心, 因此对 GIS 数据模型的认识和研究在设计 GIS 空间数据库和发展新一代 GIS 系统的过程中起着举足轻重的作用。

而在道路网建模中, 对于如何建立既可以表示道路信息, 又可以表示转弯代价和交通代价, 还要有利于计算和存储的模型这个问题, 还有待进一步探讨。

参考文献:

[1] Dao D, Rizos C, Wang J. Location - Based Services: Technical and Business Issues[J]. GPS Solutions, 2002, 6(3): 169 - 178.

[2] 赵志弘, 李志林, 余 萌. 涉及位置的信息服务: 关键趋势和商务模式[J]. 地理信息世界, 2003, 1(3): 14 - 18.

[3] 张山山, 边馥苓. 地理信息系统数据模型分析[J]. 测绘通报, 2004(8): 18 - 21.

[4] Jensen C S, Kolar J, Pedersen T B, et al. Nearest Neighbor

Queries in Road Network[C]//In: Proceedings of the 11th ACM GIS'03. New Orleans, Louisiana, USA: [s. n.], 2003: 1 - 8.

[5] Speicys L, Jensen C S, Kligys A. Computational Data Modeling for Network - Constrained Moving Objects[C]//In: Proceedings of the 11th ACM GIS'03. New Orleans, Louisiana, USA: [s. n.], 2003: 118 - 125.

[6] Karimi H A, Liu X. A Predictive Location Model for Location - Based Services[C]//In: Proceedings of the 11th ACM GIS'03. New Orleans, Louisiana, USA: [s. n.], 2003: 126 - 133.

[7] Jiang J, Han G, Chen J. Modelling Turning Restrictions in Traffic Network for Vehicle Navigation System[C]//In: Proceedings of the International Symposium on Geospatial Theory, Processing and Applications Commission IV. Ottawa, Canada: [s. n.], 2002.

[8] 石建军, 宋 延, 程世东. 车辆实时调度中的城市路网及交通描述模型[J]. 公路交通科技, 2005, 22(5): 128 - 131.

[9] 蔡先华, 王 炜, 戚浩平. 基于 GIS 的道路几何网络数据模型及其应用[J]. 测绘通报, 2005(12): 24 - 27.

[10] 王杰臣, 毛海城, 杨得志. 图的节点 - 弧段联合结构表示法及其在 GIS 最优路径选取中的应用[J]. 测绘学报, 2000, 29(1): 47 - 51.

[11] Winter S. Modeling Costs of Turns in Route Planning[J]. GeoInformatica, 2002, 6(4): 345 - 360.

[12] Feng J, Zhu Y L, Mukai N, et al. Search on Transportation Network for Location - Based Service[C]//In: Proceedings of Innovations in Applied Artificial Intelligence, Lecture Notes in Artificial Intelligence. Italia: Springer, 2005: 657 - 666.

[13] Ding Z, Güting R H. Uncertainty Management for Network Constrained Moving Objects[C]//In: Proceedings of the 15th Intl. Workshop on Database and Expert Systems Applications. Spain: Springer, 2004: 411 - 421.

[14] Ding Z, Güting R H. Managing Moving Objects on Dynamic Transportation Networks[C]//In: Proceedings of the 16th International Conference on Scientific and Statistical Database Management. Washington, DC, USA: IEEE Computer Society, 2004: 287 - 296.

[15] Güting R H, de Almeida V T, Ding Z. Modeling and Querying Moving Objects in Networks [R/OL]. Informatik - Report 308, Fernuniversität Hagen, 2004. <http://www.informatik.fernuni-hagen.de/import/pi4/papers/PaperMon.pdf>.

(上接第 202 页)

[3] Johansen B. Windows 应用高级编程[M]. 北京: 清华大学出版社, 2003.

[4] Microsoft Corporation. 智能客户端 Offline Application Block [EB/OL]. 2004. <http://www.microsoft.com/china/msdn/>

[library/architecture/architecture/architecturetopic / SCOfflineAppBlockcover.msp](http://www.informatik.fernuni-hagen.de/import/pi4/papers/PaperMon.pdf).

[5] 胡海璐. XML Web Services 高级编程范例[M]. 北京: 电子工业出版社, 2003.