

# 移动数据库 SwiftDB 的嵌入式 GIS 设计与实现

沈洁, 徐立臻

(东南大学计算机科学与工程学院, 南京 210096)

**摘要:** 结合东南大学计算机系数据库及信息系统研究室自行开发的移动数据库系统 SwiftDB, 提出了嵌入式 GIS 的实现方案。由于嵌入式设备的内存和运算速度无法与 PC 相比, 因此提出了矢量栅格存储结构, 在此基础上设计了动态划分地图区域的方法。该结构在最短路径查询中获得了较好的结果。由于移动环境具有低带宽、高延迟、易中断的特点, 因此根据所传输的数据量, 提出了不同的地图数据更新方法, 在维护数据一致性的同时也提高了系统的效率。

**关键词:** 移动数据库; 最短路径查询; 地图更新

## Embedded GIS Design and Implementation Based on Mobile Database SwiftDB

SHEN Jie, XU Li-zhen

(Department of Computer Science and Engineering, Southeast University, Nanjing 210096)

**【Abstract】** Combined with the mobile database system SwiftDB developed in the Laboratory of Database and Information System of Southeast University, the implementation method of embedded GIS is brought forward. Because the computing and storage capacity of embedded equipment are both less than PC, a data structure of vector and grid combination are presented. Based on this structure, an idea of dynamic block division is designed. The data structure has been applied in the example of shortest path query. Due to low bandwidth, high delay and easy to interrupt in the mobile environment, this system designs different methods for updating the map data based on the data volume. These methods benefit the consistency of data and the performance of the whole system.

**【Key words】** mobile database; shortest path query; map update

随着移动数据库技术从研究领域向应用领域发展, 各种基于移动数据库的产品也在陆续出现, 如 Microsoft SQL Server 2000 Windows CE Edition, Oracle Lite, Sybase UltraLite, 人民大学的“小精灵”等。这给开发嵌入式 GIS 系统提供了很大的便利性。

SwiftDB是由东南大学计算机系数据库及信息系统研究室开发的移动数据库系统, 它不仅在移动终端实现了一个高效的嵌入式数据库系统EDBMS, 还通过同步服务器较好地解决了移动事务的处理与数据一致性问题。SwiftDB的中心数据库是PC上的数据库, 它控制和维护所有的数据, 具有全局性, 可以是任何当前主流的企业数据库。同步服务器通过ODBC连接中心数据库, 负责向移动用户提供数据订阅服务, 处理移动事务并维护数据的一致性。移动终端的用户向同步服务器端订阅自己感兴趣的数据并下载到本地EDBMS, 而EDBMS可以自由地管理本地数据, 执行查询或移动事务, 也可以在任何时候向同步服务器发送同步请求, 从而与中心数据库中的数据达到一致。本文提出了一种在SwiftDB的基础上实现嵌入式GIS的解决方案<sup>[1]</sup>。

### 1 基于 SwiftDB 的嵌入式 GIS 的数据组织结构

#### 1.1 地图信息存储结构

表达地图信息有两种基本方法<sup>[2]</sup>: (1)栅格数据结构, 即把空间分成一系列栅格单元, 每个单元代表有限但确定的地球表面, 单元可以具有任何几何形状, 只要能组合出表达研究区域的面即可。这种数据结构的特点是属性明显、定位隐

含;(2)矢量数据结构。它用一系列X,Y坐标作为位置标志符, 用于描述各种地理要素, 能够精确定位空间位置。这种数据结构地特点是定位明显、属性隐含。考虑到本系统运行在嵌入式环境下, 受到存储空间的限制, 引出block和区域的概念, 提出了一种适合于嵌入式系统的栅矢地图数据组织。本系统地图的背景图片采用JPG格式。由于JPG图片没有任何标识信息, 因此为了实现定位必须对图片进行手动标注。通过给出屏幕地图坐标系中的任两点 $P1(X1, Y1)$ 、 $P2(X2, Y2)$ 的精确的经纬度 $(1, 1)$ 和 $(2, 2)$ , 就可以得到地图上任意一个点 $(X, Y)$ 的经纬度 $(, )$ :

$$= 1+(2-1) \times (X-X1) \div (X2-X1)$$

$$= 1+(2-1) \times (Y-Y1) \div (Y2-Y1)$$

JPG 图片经过标注后可作为电子地图存入数据库, 其相应的经纬度信息是关联地图数据和空间数据的重要纽带。用户可根据自身的需要从中心数据库中下载相应的电子地图。

为了提高从图形到数据库查询的速度, 将地图按栅格方式划分为若干大小相等的基本块(见图1), 基本块是图形定位的最小单元, 根据用户的需要可以将若干基本块组合为区域。若一电子地图的经度跨度为 $(1, 2)$ , 纬度跨度为 $(1, 2)$ , 每个基本块的经度跨度为, 纬度跨度为, 地图从左上角开始从左至右从上至下进行编号, 如图1所示。某时刻 GPS

**作者简介:** 沈洁(1982-), 女, 硕士研究生, 主研方向: 移动数据库和嵌入式 GIS; 徐立臻, 教授

**收稿日期:** 2006-08-30 **E-mail:** sidaimien@yahoo.com.cn

接收机送来的经纬度数据为( , )，此数据所在的基本块号为

$$\left\lfloor \frac{\beta - \beta_2}{\Delta\beta} \right\rfloor \times \frac{\alpha_2 - \alpha_1}{\Delta\alpha} + \frac{\alpha - \alpha_1}{\Delta\alpha}$$

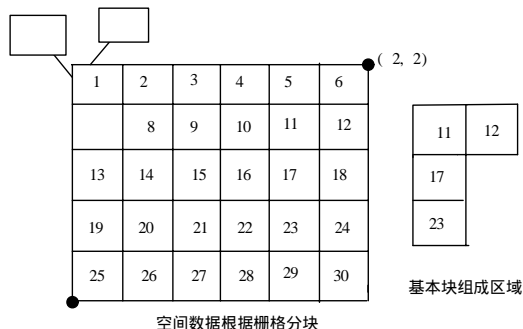


图1 基本块结构及区域划分

## 1.2 图层信息存储结构

由于嵌入式设备本身内存容量的限制，采用将图元素分层分类存放的方法，如图2所示。每个图层元素的特征与数据逻辑关系不同，因此，组成该图层的区域的大小与组合方式也不同。由于不同的图层有不同的区域划分方式，因此同一基本块在不同图层可能属于不同区域。根据不同图层的需要灵活地组织基本块，形成符合该图层的区域划分。

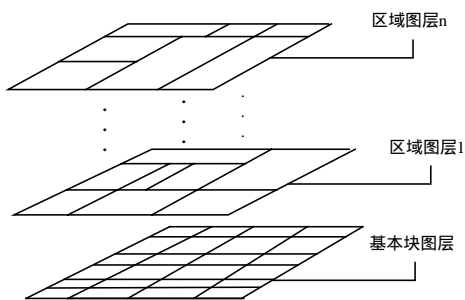


图2 图元素的分层分类

在移动数据库 SwiftDB 中，图层信息可以表示为图层数据表的形式，其结构如表1所示。

表1 图层数据

字段名	字段类型
lay_no 图层号	Integer
lay_type 图层类型	Integer
lay_nam 图层名称	Char
spat_table_nam 空间数据表名	Char
pro_table_nam 属性表名	Char
index_table_nam 空间索引表名	Char
Leftx 图层左上角 X 坐标	Double
lefty 图层左上角 Y 坐标	Double
rightx 图层右下角 X 坐标	Double
righty 图层右下角 Y 坐标	Double

将图元素分层分类存放的方法可利用相关数据的聚集提高查询速度，并能够根据算法的需要灵活的组织空间数据。

## 2 基于 SwiftDB 嵌入式 GIS 的功能设计

### 2.1 实时监控功能

实时监控的含义是指用户在任意时刻通过发出定位查询指令来获得运输工具所在的地理位置(经度、纬度、速度等信息)并在电子地图上显示出来。目前，移动定位主要有两种方式：GPS定位和GSM定位。GPS具有性能好、精度高、应用

广的特点，是迄今最好的导航定位系统。GSM接收机比GPS接收机简单、直接，费用低，用户通过接收机内置设备发送手机短消息，确认自身所在位置，但不能随机定位。比较适合对实时定位要求低的监控系统。鉴于上述原因，本系统采用GPS定位方式<sup>[4]</sup>。由于GPS接收机每秒钟均有数据输出，产生的数据量大，特别是在移动设备存储空间有限情况之下，冗余数据需要专门的判断和处理。判断过程如下：

(1)如果读取GPS输出时间间隔比较长，每次读取的数据超过缓冲区的大小则丢弃。

(2)对已经读到了缓冲区的数据，可能存在多条有效报文，只需读取第1条有效报文，其余的报文数据丢弃。

(3)若当前GPRMC报文的速率不为0则GPS一定在运动中，当前数据不是冗余数据，若速率为0且当前读取经纬度与最近保存点的经纬度相等，则可判断GPS静止且当前数据冗余，可舍去。

根据GPS获取的数据，需要在地图上显示相应的轨迹，这时启用轨迹推算算法，步骤如下：

(1)已知保存在数组中的最后一点坐标为 $P_n(X_n, Y_n)$ ；

(2)在数据库找到该点所处的BLOCK；

(3)找到该BLOCK所包含的道路信息，如某道路线段的端点为 $P_i(X_i, Y_i), P_j(X_j, Y_j)$ 则根据点与线的求交公式 $d^2 = (X_n - X_i)^2 + (Y_n - Y_i)^2 + [(X_j - X_i) + (Y_j - Y_i)]^2 / [(X_j - X_i)^2 + (Y_j - Y_i)^2]$ ，当 $d^2 < 2$ 并且 $(X_n - X_i)$ 与 $(X_n - X_j)$ 反号时，点 $P_n$ 在该线段有效区间，认为目前该点正在该道路上；

(4)根据该道路线段的系数 $K$ 推算出轨迹点 $P_{n+1}(X_{n+1}, Y_{n+1})$ (其中，系数 $K$ 为道路线段的斜率)；

(5)判断 $P_{n+1}$ 是否已经超出概线段的范围。如果未超出范围，则将该点记入数组并标记该点为推算点；

(6)否则说明此时轨迹已经行驶到某道路口，下一点的预测必须要根据接收到的正确的GPS信号后做出判断。

(7)当中断后第1次接到GPS正确信息后，重复步骤(1)~步骤(3)，找到该点所属的道路信息。并在已经保存在数组中的依次找出标注为推算的点，所属的道路信息，如遇到与此正确点所属的不为同一道路，则从数据库找出这两条道路的交点插入数组中。

### 2.2 路线寻优功能

寻找两点最短路径公认的最好算法为Dijkstra算法，但是由于经典Dijkstra算法在搜索最小值和修改最短路径长度上会耗费大量的时间，从而造成整个算法的时间复杂度为 $O(n^2)$ ，非常不适合于嵌入式设备。因此，笔者认为就嵌入式应用而言必须对经典Dijkstra算法进行优化。

经典Dijkstra算法一种有效的优化方法：减小算法中成功搜索的搜索范围，以尽快到达目标结点<sup>[5]</sup>。由于本系统根据配送点的分布情况对物流配送图层进行区域划分，因此可以采用如下算法减少空间分析时的数据量。

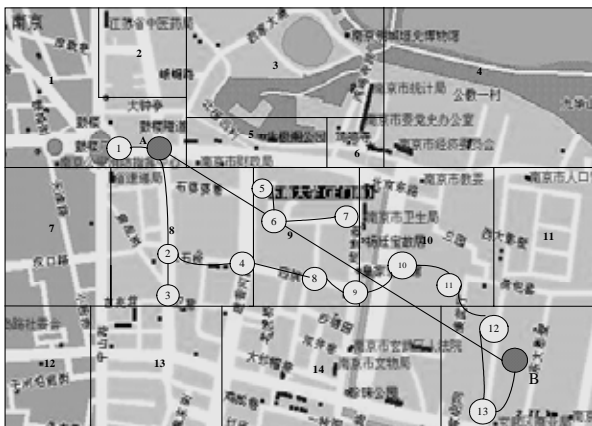
(1)判断起始点和终点所在的区域，如果两点在同一区域内，则提取该区域内主要道路信息，算法结束；否则转(2)。

(2)将起始点和终点连接起来作为直线，判断该直线的方向，并计算在该方向上通过的区域，若区域数>某个阈值，转(3)，否则提取区域内的道路信息。

(3)提取区域内道路的倾斜角度，并与该直线的方向进行比较，若二者角度的差值 $< 45^\circ$ ，则提取该道路信息；否则舍去该道路信息。图3中起点和终点分别为A和B，将它们用直线连接起来，可以发现从A到B经过1号、8号、9号、10

号和 15 号区域,由于经过的区域较多,需要判断区域中各条道路与直线方向之间的夹角,然后提取相应的道路信息,本例中为图 3 中所标注的弧线。

图 3 区域划分在最短路径查询中的应用



此方法使搜索方向智能地趋向于目标结点,减少了算法中遍历的结点个数,从而提高了搜索速度。由于根据方向性来判断查找的区域,因此最先找到的路径基本上为最优的,如果在初始时所确定的区域内没有找到最短路径,则算法的最坏情况是退化为对所有区域的查找。

### 2.3 地图数据同步功能

由于嵌入式设备存储容量的限制,不能够将某个城市所有的地图信息全部存放在嵌入式设备中,而是采用根据应用需求分批下载的策略。在移动环境里,网络的带宽存在着不对称性,下行带宽(服务器到移动设备)大于上行带宽(移动设备到服务器),移动设备甚至可以在不能发送信息的情况下接受服务器发给它的信息;移动设备在网络里是间歇性入网,会频繁地断接;同时移动设备本身由电池供电,供电时间相当有限<sup>[6]</sup>。因此,本系统根据不同的情况设计了两种地图数据的同步方式。

(1)对于图形文件,如果采用一次传输的方式,需要耗费比传输一般数据更多的时间,因此,应当改进现有的图形传输方式。在中心数据库端,将某张尺寸较大的背景图分成若干小块,并对分成的图块进行编号。当移动端请求传输图片

时,按照图块的编号顺序分块的对图片进行传输,如果在传输过程中网络中断,那么在移动端再次入网后,可以从网络中断时正在传输的图块开始继续往后传输。实现这种传输方式的方法是采用 MSMQ 消息队列技术,在断接的情况下,消息传递系统将要发送的消息立即保存到消息队列,客户机可以在网络可用的时候再发送队列中保存的消息。当网络一旦建立连接,系统就试图把没有发送的消息尽力重新传输出去。因此 MSMQ 技术能够更有效地利用网络带宽。

(2)对于地图数据,可以利用移动数据库 SwiftDB 提供的增量下载的方式进行更新。在执行地图数据的下载时,由订阅数据时自动生成的触发器和存储过程跟踪被订阅数据的变化,检查同步冲突,然后检查相应的数据库所订阅的数据,将变化的部分增量下载到相应的表中,使它重新与中心数据库取得一致。

### 3 结论及展望

使用基于移动数据库 SwiftDB 的嵌入式 GIS 时,移动客户端可以在移动时存取后台数据库并且可以带着后台数据库副本移动,即使在网络通信断开的情况下,仍然可以根据移动端存储的数据使用 GIS 功能。而在移动和频繁断接条件下的地图数据更新则交由系统根据不同的情况加以完成。这些特性是本地不带数据的嵌入式 GIS 所无法完成的。但是,对通信频繁断接操作的支持、地图数据的查询优化以及如何提高有限资源的利用率和系统效率等问题仍是今后需要解决和完善的地方。

#### 参考文献

- 徐立臻,张洁. SwiftDB: 一个支持行业应用的移动数据库系统[J]. 应用科学学报, 2004, 22(4): 553-556.
- 欧阳骞. 基于嵌入式的移动地理信息系统研究与设计[D]. 南京: 东南大学, 2005.
- 熊庆文,边馥苓. 基于嵌入式数据库系统的移动 GIS 应用体系结构研究[J]. 武汉大学学报, 2006, 31(1): 86-89.
- 田根,童小华,张锦. 基于 PDA 的 GIS 系统研究与开发[J]. 计算机工程, 2004, 30(2): 76-85.
- 王开义. GIS 领域最短路径搜索问题的一种高效实现[J]. 中国图象图形学报, 2003, 8(8): 951-956.
- 王珊. 移动数据库及其应用[J]. 计算机应用, 2000, 20(9): 1-4.

(上接第 46 页)

本文主要研究了基于模糊理论的信息系统风险计算方法。阐述了影响风险计算的数据,讨论了影响风险计算的函数因子,用层次分析法确定因子的权重,描述了因子权重确定的模型和流程,给出了风险计算量化的方法、流程和具体计算实例。

由于不同的信息系统有其不同的特性,确定风险计算的函数因子也会各有不同,本文只针对影响风险计算的因素划分为几个方向,从而确定二层次模糊综合评估模型,在实际工作中可结合具体的信息系统确定多层次的评估模型,尽可能客观、准确地描述信息系统所面临的风险。

#### 参考文献

- 钱刚. 信息系统安全管理[M]. 南京: 东南大学出版社, 2004.
- 计算机信息系统安全等级保护管理评估——风险分析与评估第 1 部分: 安全风险评估程序(征求意见稿)[EB/OL]. (2006-03-10). <http://www.xfocus.net>.
- 王兴隆. 空管自动化系统软件可靠性评估和预测方法的研究[D]. 天津: 中国民用航空学院, 2006.
- 模糊综合评判方法[EB/OL]. (2006-03-10). <http://zhjyx.hfjy.net.cn/>.
- 金菊良,魏一鸣,丁晶. 基于改进层次分析法的模糊综合评价模型[J]. 水利学报, 2004, 49(3): 65-68.