

以版本控制的遥感解译目标数据集管理系统

王培忠, 严卫东, 边辉, 孙彬, 马心璐

(西北核技术研究所, 西安 710024)

摘要: 为解决卫星监测目标解译过程中“长事务处理”及多用户编辑带来的问题, 实现对监测目标多个时刻状态的有效存储与管理, 支撑卫星监测任务。首先, 分析了 ArcSDE 的版本控制原理, 研究其在卫星监测目标管理中的应用可行性; 其次, 在对卫星监测目标管理及任务需求分析的基础上, 设计出基于 ArcSDE 版本技术的卫星监测目标存储、管理平台框架, 主要包括目标解译过程中各解译版本对比、历史回溯及冲突协调处理等核心功能模块, 实现对卫星监测目标各解译结果之间的访问查看、对比及目标历史状态的再现; 最后, 利用 ArcGIS Engine 进行二次开发, 通过对系统主要功能的实现与验证表明, 系统能有效提供卫星监测目标历史重构、跟踪分布等支撑。

关键词: 版本管理; ArcSDE; 卫星监测; 历史回溯

DOI: 10.3724/SP.J.1047.2012.00442

1 引言

卫星监测目标大多是以空间要素的形式存储在关系型数据库中^[1]。由于关系数据库多采用“锁定→修改→释放”的策略, 实现对多用户并发操作的控制, 不太适合对卫星监测目标数据库的并发访问。因为卫星监测目标的解译、编辑过程具有反复性及长期性, 需要的时间可能在几个月甚至更长, 这种情形被称为“长事务处理”^[2]。解决“长事务处理”问题的一个有效方法是利用版本管理机制, 存储与管理卫星监测目标多个时刻的状态, 通过进行历史对比, 来提高卫星监测的科学性及置信度。

ArcSDE 版本技术也能够很好地解决“长事务处理”及多用户同时编辑的问题。利用版本化, 多个用户可对 ArcSDE 地理数据库中的同一目标进行编辑, 而无需应用锁。这样可隔离用户在多个编辑会话中的工作, 使得用户进行编辑时不必锁定生产版本中的监测目标或直接影响到其他用户, 且无需备份数据。

传统卫星目标解译、监测研究中, 目标监测过程的中间状态很难被保存, 无法实现对历史目标回溯; 而且不同研究人员对同一目标的解译无法实现结果的自动对比分析; 该系统以版本技术, 完成了历史状态重建、跟踪变化, 统计分析和趋势分析等

功能。解决了监测目标历史回溯及目标对比分析困难的问题。

2 ArcSDE 版本控制原理

ArcSDE 版本是整个数据的快照, 而不是数据库的硬拷贝, 它只保存数据库中变化的内容, 对于没有更新变化内容, 只物理存储一次。利用版本化, 多个用户可对 ArcSDE 地理数据库中的同一监测目标进行编辑, 而无需应用锁。这样可隔离用户在多个编辑会话中的工作, 使得用户进行编辑时不必锁定生产版本中的监测目标或直接影响到其他用户, 且无需备份数据^[3]。利用版本化, 可以将监测目标解译过程中的多个状态进行保存, 实现历史重构或历史对比监测, 以提高监测准确度。

每个 ArcSDE 地理数据库都具有一个被称为 DEFAULT 的默认版本; 与其他版本不同, DEFAULT 版本始终存在, 且不能被删除。DEFAULT 版本为根版本, 它是其他所有版本的祖先版本, DEFAULT 保存了数据库当前的状态, 创建版本时所创建的是地理数据库的某种“视图”。无论创建多少个版本, 各表和要素类将仅在数据库中存储一次, 数据库所有的修改均保存在增量表中。通过版本可以编辑版本化数据并查看所作修改, 当

收稿日期: 2012-04-07; **修回日期:** 2012-07-16.

作者简介: 王培忠(1981-), 男, 硕士, 工程师, 主要研究方向: 遥感解译及 GIS 应用开发。E-mail: myhome8556@126.com

将不同的版本所作修改提交到父版本时,会产生冲突,通过版本协调技术实现冲突解决,各子版本之间通过父版本协调冲突,而不是直接进行冲突协调^[4-5]。

3 系统及核心功能模块设计与实现

3.1 卫星监测目标管理特点

卫星监测目标管理系统是卫星监测任务的技术支撑,是卫星监测工作不可或缺的部分。用户通过卫星图像解译获取了大量的监测目标信息。系统要为用户提供大量目标存储、管理平台,并提供解译目标矢量入库、更新、对比、历史追踪等业务。卫星监测目标管理具有如下特点:

(1)目标解译长时间性:地面目标不总是一直处于停滞状态,一系列的变化表现在序列卫星图像上就是目标从无到有,再到发展变化的一系列遥感特征,通过遥感特征分析、解译目标是一个相对长过程,在分析、判别过程中又需要记录各时刻的状态,一方面是为了将各状态进行对比分析,另一方面是为了对过去某一历史状态进行回溯。

(2)目标解译反复性:简单的目标可以通过解译直接定性,给出结论,复杂目标的分析过程可能充满反复性,需要不断重复“判别—验证—修改—结论”的过程,这就需要系统保持目标的现势性,同时,还要记录其历史状态。

(3)多用户操作:卫星监测目标的解译具有不确定性,尤其是地面活动的分析解译,对同一目标及活动的分析、判读通常需要多人共同参与,然后将结果作对比分析。这就需要系统能够支持多人同时协同工作,且能保证目标数据的完整一致性。

针对上述业务特点,充分利用版本控制技术,实现对卫星监测目标的存储及管理,可有效支持卫星监测工作。

3.2 系统功能设计

目标旨在将遥感解译结果及各种信息资料管理集成于一体,系统采用 ArcSDE 加 Oracle 来贮存和管理典型目标。并且需要反复对比分析的过程,同时,目标本身也是变化的,因此,解译过程中需要记录目标多个时刻的状态,以便多个状态之间进行对比和历史回溯。卫星监测目标解译、存储、管理业务流程如图 1 所示。

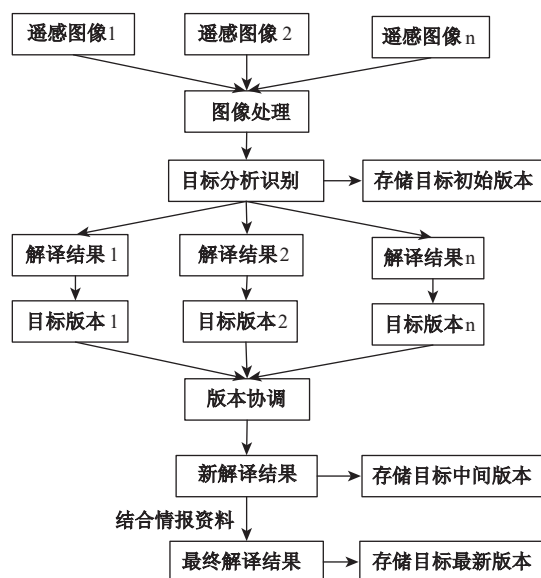


图 1 卫星监测目标解译、管理流程

Fig. 1 Flow chart of target interpreting, storing and managing

根据目标管理及卫星监测任务需求,将卫星监测目标管理系统的功能设计为:

(1)创建和删除目标监测版本:一旦启动监测任务,即为目标创建解译版本,目标状态的每次变化均对应新版本,用户可以实现对版本的创建和删除。

(2)查看版本信息:用户可以浏览所有的版本信息,包括版本的创建时间、创建者、权限及父子版本等相关信息。

(3)目标历史状态回溯:用户可以查看不同版本的目标,也可以回溯到某一版本的某一时刻,以查看目标不同时刻的状态。

(4)监测目标版本对比:以图形化的形式对比、检测出两个版本之间的差异情况,这种差异有或者是两个时刻之间的变化,或者是对同一目标不同监测人员解译结果之间的差异,也可以通过版本对比查看某个时刻的历史数据库状态。

(5)版本冲突的协调及合并:利用版本冲突发现不同解译人员对同一目标解译结果的差异,经过综合分析确定最终可信的结果,通过版本的协调及合并实现最终结果提交入库。

3.3 监测目标版本管理树状结构及遍历设计

当 ArcSDE 服务器的一个实例第一次启动时,就建立了数据库缺省的状态和版本——SDE.DEFAULT,它是服务器中所有版本的父版本,因此,

系统始终会为地理数据库启用版本化。与其他版本不同, DEFAULT 版本始终存在, 且不能删除。在大多数工作中, 它是数据库的发布版本, 表示系统的当前状态, 可以将其他版本中的变更提交到 DEFAULT 版本, 从而逐步维护和更新 DEFAULT 版本^[6]。此外, 还可以像编辑其他版本一样, 对 DEFAULT 版本直接进行编辑。版本的创建、删除、查看等操作可以通过 ArcGIS 桌面或者调用 ArcGIS Engine 中的 IVersion 接口实现。

整个 SDE 中的所有版本构成一个树状结构。在 SDE 中一般只允许你删除叶节点版本。如果想删除某个版本, 首先, 需要生成该版本及其所有的子版本的版本树, 然后, 遍历该版本树, 得到遍历后各个版本的排列名, 而后按从子版本到父亲版本的顺序删除就可以了, 需要注意的是只有该版本的用户才有权限删除它^[7-8]。

3.4 历史版本回溯及浏览的设计与实现

从祖父版本 DEFAULT 衍生的所有子版本都保存在 SDE 中, 版本之间的对比浏览通过以下的方法实现: (1) 连接数据库中版本 1, 通过 IVersionedWorkspace、IFeatureClass 等接口实现版本 1 中目标图层的读取、显示, 同理, 读取、显示版本 2 中的监测目标图层; (2) 访问增量表中的变化信息, 根据变化信息唯一 ID 访问对应版本中的监测目标, 并以不同的标志进行标记, 从而实现两个版本中差异信息的查看; (3) 加载不同版本的目标图层至同一窗口中, 实现不同版本的浏览、对比查看。

在实际工作中, 经常需要追踪某个监测目标的历史变化过程, 回溯某个时刻的状态。而数据库中保存的总是版本的最新状态, 为了达到历史回溯的目的, 需要记录数据库每次修改前后的状态; 在数据库中, 每个目标(即每个空间要素)对应表的一条记录。由于每次目标的更新总是少量的, 因此, 只需保存变化信息, 而无须保存所有的东西, 通过访问数据库的变化信息来实现历史回溯^[9]。鉴此, 我们采取下面的策略:

建立地理数据库存档, 它是记录、管理和分析数据更改的一种机制, 能够记录卫星目标在某一时刻特定属性值以及随时间如何变化的, 通过版本化的历史存档记录, 用户可以访问已版本化的地理数据库中的目标的变化情况。

建立目标数据集版本化存档后, 以 SDE.DEFAULT 数据结构建立一个存档类, 增加字段 GDB_FROM_DATE 和 GDB_TO_DATE 来保存目标存档更改的时间。当启用存档时, 目标 DEFAULT 版本的所有行都将按同一时间戳被复制到存档类。所有行的 GDB_FROM_DATE 属性都带有启用存档操作的日期和时间的的时间戳。所有行的 GDB_TO_DATE 属性都带有 12/31/9999 时间戳。无论何时只要一个属性具有 12/31/9999, 它都是 DEFAULT 版本中对象的当前表示。将编辑内容保存或提交到 DEFAULT 版本后, 地理数据库会自动将更改存档到存档类。

目标的历史回溯通过归档类来实现, 归档类记录数据库中目标的增加、删除和更新, 其策略如下: 当插入新目标并将编辑内容提交到 DEFAULT 版本时, 会在存档类中插入新行, 同时, 通过更新提交操作的时间戳, 新行的 GDB_TO_DATE 属性显示 12/31/9999, 因为还没有更新或删除该要素; 当更新要素时, 会通过存档操作的时间戳设置 GDB_TO_DATE, 并插入新行以显示目标的当前表示。此新行的 GDB_FROM_DATE 通过存档操作的时间进行设置, 而且仍将显示 12/31/9999, 因为还没有修改或删除该行; 当删除要素时, GDB_TO_DATE 将通过存档操作的时间戳来更新。要回溯目标某一时刻的状态, 通过访问归档类的 GDB_FROM_DATE 和 GDB_TO_DATE 属性来实现。历史回溯流程如图 2 所示。

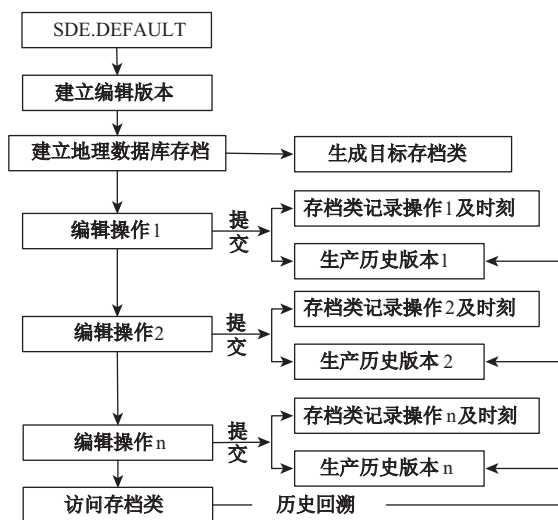


图2 目标历史版本回溯

Fig. 2 Tracing the historical version of the targets

图 3 所示为目标数据库归档类中的某些目标在 2011 年 5 月 19 号、6 月 28 号等几个时刻的编辑操作,包括目标删除、移动、修改属性及大小等,归档类记录了目标的变化信息。图中褐红色部分为未变化目标,这些目标的 GDB_TO_DATE 属性值保持初始化时间“9999-12-31 23:59:59”,通过历史版本回溯可以容易地查看某个时刻某支持中心的目标状态,图 4 所示为 2011 年 3 月 2 号 10:23:21 时刻的目标状态。

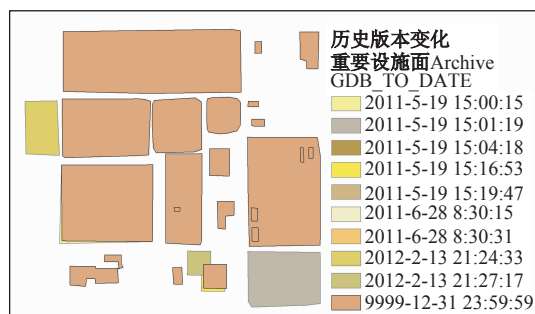


图 3 记录变化信息的归档类

Fig. 3 Records of changed targets in the archive class



图 4 历史浏览图(2011/3/2 10:23:21 时刻)

Fig. 4 The history browse view
(Time: 2011-03-02, 10:23:21)

3.5 版本合并的设计和实现

当多个用户对同一目标进行监测时,会建立多个编辑版本,用户在新的编辑版本上进行目标分析、解译工作。最后,需要将多个用户的识别结果进行综合对比分析,给出可靠结果,其需提交到父版本中,作为某一时刻的目标状态信息。由于存在多用户对同一目标进行编辑,在提交更新结果至父版本时可能发生冲突,冲突发生在多个用户对同一目标进行不同的更新时。下面详细分析冲突发生的过程,假设从 T0 时刻创建版本,经过编辑工作到

达 T1 时刻,再经过一段时间的编辑工作到达 T2 时刻^[10]。

T0 时刻:用户 Ua 和 Ub 分别从已有版本 V 上创建编辑版本 A 和 B,此时版本 V、A、B 的状态是一致的。

T1 时刻:用户 Ua 在版本 A 中进行目标编辑后,提交更新结果到版本 V,而用户 Ub 并不知道,他继续在版本 B 上进行解译、编辑工作。

T2 时刻:用户 Ub 完成对版本 B 的解译工作后,保存并提交更新到版本 V 中。而此时版本 V 已经被 Ua 更新过了。如果 Ub 在版本 B 中对 Ua 更新过的要素也进行了修改,那么冲突不可避免地发生了。

完成版本编辑操作后,可以将更改结果合并到该版本的任意原型版本(如父版本或 DEFAULT 版本)。要实现版本合并,必须协调所有版本冲突,然后发布协调结果^[11-12]。为了后面表述方便,在此将版本 V 称之为目标版本,版本 A 和 B 称之为编辑版本,将 T0 时刻的版本 V、A、B 称之为冲突前版本。在版本协调的过程中,必须首先检查冲突:

(1)在当前正在编辑的版本和目标版本中对同一要素进行更新。

(2)在当前正在编辑的版本和目标版本中对同一要素进行更新。

(3)在当前正编辑的版本和目标版本中修改与要素或关系类关联的拓扑。

在解决版本冲突时,需要决定以编辑版本还是以目标版本进行更新,如果选择的解决规则是支持目标版本,则当前编辑会话中所有冲突要素均被替换为它们在目标版本中的制图表达;如果选择的解决规则是支持编辑版本,则当前编辑会话中的所有冲突要素均优先于目标版本中的冲突制图表达。在版本冲突处理窗口,用户可以查看每个冲突空间要素在各个版本中的属性值和几何图形,可以指定以哪个版本的值来解决冲突。图 5 所示为版本冲突信息,图 5(a)为冲突目标及属性信息,图 5(b)为目标冲突显示,绿色框内目标为冲突前版本,红色为冲突版本,蓝色目标原始状态。完成版本协调后即可进行提交,更新目标到最新状态。版本协调的流程如图 6 所示。

利用 ArcGIS Engine 进行二次开发, ArcGIS Engine 提供了大量的版本操作及目标要素类操作接口。通过调用 ArcGIS Engine 中 IVersionEdit、

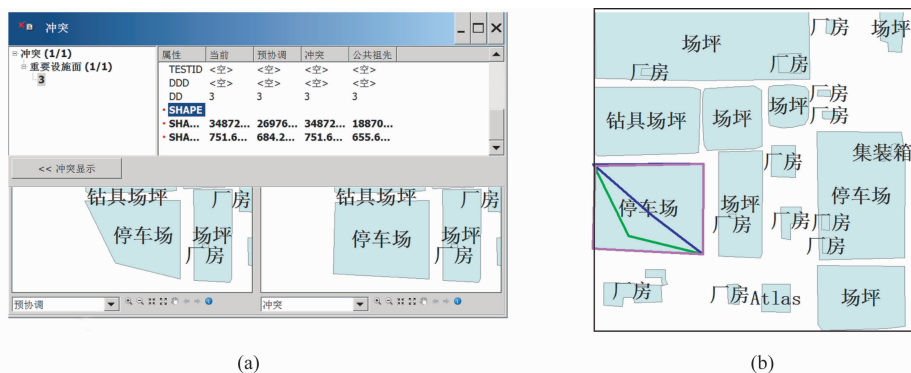


图5 目标版本冲突信息

Fig. 5 The conflicting information between versions

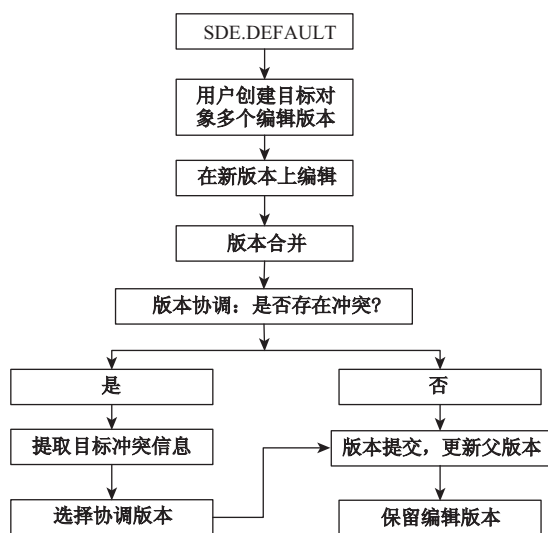


图6 版本协调流程图

Fig. 6 Flow chart of version reconciling

IVersion、IConflictClass、IVersionedWorkspace 等接口及类实现了版本的查看、历史回溯、冲突协调解决等功能;通过调用 IFertureWorkspace、IFeaturerelayer、IFeatureClass 等接口来编辑修改目标在版本中的值,实现对目标的版本化编辑^[13]。

4 结论

(1)利用 ArcSDE 的版本数据模型,本系统能完整地跟踪与记录某一具体卫星监测目标在整个生命周期的历史空间变化过程及专题属性信息^[14];以 ArcGIS Engine 二次开发技术,实现了卫星监测目标的管理功能。结果表明:系统运行稳定,达到了预期的效果。ArcSDE 最新版本精简、优化了系统表,使得在管理频繁的编辑、访问大量

的历史信息时效率更高,将最新版本技术应用到卫星监测目标管理系统中对提高目标对象存储、管理和跟踪分析能力非常重要,这是系统今后改进的方向。

(2)卫星监测目标管理系统扩展了 GIS 应用范围,系统在卫星监测目标历史重构、跟踪分布、变化等方面具有一定的优势^[15]。实践证明,ArcSDE 版本支撑下的 GIS 系统能很好地应用于较多行业的卫星监测、规划管理等领域。

参考文献:

- [1] 严卫东,赵亦工,李真富,等. 地下核爆炸卫星监测关键技术[J]. 光电工程,2009,36(8):67-73.
- [2] 高仇生,赵英俊. 卫星遥感技术在核保障监督中的应用现状分析[J]. 世界核地质科学,2008,25(4):231-236.
- [3] 韩敏,赵松龄. 基于特征-版本的时空数据模型在湿地管理中的应用研究[J]. 测绘科学,2007,32(6):140-144.
- [4] 李伟,刘仁义,刘南. 基于任务划分和多版本技术的 GIS 空间数据协同处理研究[J]. 浙江大学学报(理学版),2005,32(4):475-480.
- [5] 程承旗,胡雪莲,马嵩乃. COGIS 概念及体系结构研究[J]. 地理信息世界,2003,1(1):23-29.
- [6] Churcher N, Churcher C, Grouparc A. Collaborative approach to GIS [A]. 8th Annual Colloquium of the Spatial Information Research Centre, 1996:156-163.
- [7] 何保国,朱国宾. 基于版本管理的多用户并发编辑控制策略研究[J]. 地理空间信息,2007,5(4):1-4.
- [8] 郑海鹰,张立朝,孙婷. 基于 Geodatabases 的数据复制技术研究[J]. 测绘科学技术学报,2007,24(3):175-178.
- [9] Langran G. A review of temporal database research and its use in GIS applications[J]. International Journal of Geographic Information Systems, 1989, 3(3): 215-

- 232.
- [10] 夏宇,朱欣焰,吕维. 基于 ArcSDE 的空间数据版本管理问题研究[J]. 计算机工程与应用,2007,43(13):14-16.
- [11] 唐嘉锴,唐代生,胡觉. 时态 GIS 及版本管理原理在森林资源数据更新中的应用研究[J]. 中南林业调查规划,2008,27(1):46-49.
- [12] 郭继发,刘臻,崔伟宏. 基于时空特征分析技术的烤烟 GIS 系统[J]. 地球信息科学学报,2009,11(3):370-375
- [13] 操震洲,李清泉. 基于 ArcSDE 和 ArcGIS Engine 的版本管理系统的设计和实现[J]. 测绘与空间地理信息,2006,29(1):76-79.
- [14] 舒飞跃,闫国年,陆婧,等. 基于知识对象的土地管理空间数据库模型设计与实现[J]. 地球信息科学学报,2010,12(3):348-357.
- [15] 杨忠德,曹新平. 基于时空关联的警用信息系统的设计与应用[J]. 地球信息科学学报,2011,13(4):480-485.

System of Dataset of Remote Sensing Interpreting Target Based on Version Management

WANG Peizhong, YAN Weidong, BIAN Hui, SUN Bin and MA Xinlu

(Northwest Institute of Nuclear Technology, Xi'an 710024, China)

Abstract: Satellite monitoring is a mean of verifying the suspicious activities by interpreting the state and analyzing the targets in satellite imagery combining other information, management of targets of satellite monitoring usually has three characteristics, the process of monitoring targets takes a long time, is a repetitious one, and needs multi-users' editing. In practical study the targets are usually stored in a relational database in the form of spatial features, and a "lock→change→release" strategy is usually applied to relational database. However this approach is not suitable for concurrent accessing database. To resolve the problem, so as to restore and manage much status of monitoring targets in the lifecycle of remote sense interpretation, and to support the mission of satellite monitoring, firstly, a discussion was made out about the mechanism of ArcSDE version, and feasibility was researched on that the version was applied to managing satellite monitoring targets, by the version technology the users were allowed to edit targets without locking features, and the other user's edited work was not affected. Therefore, Version technology can resolve the problem caused by multi-user concurrent operation and long transaction processing; Secondly, the feature and mask of satellite monitoring targets was deeply analyzed, it was concluded that the management of targets of satellite monitoring was closely relate to the version management technology. and then the framework of target managing system of Satellite monitoring based on ArcSDE version is designed basing the aforementioned study, the system includes the main functions of version contrasting, historical version tracing, and conflict-resolving strategy among versions; using the system the result of targets interpretation and reappearance of historical status of targets could be accessed and viewed. At last the system is built by the ArcGIS Engine component. The result of system indicated that this system showed good performance in viewing various status of targets and contrasting the same target among various results of interpreting targets, as well as recalling of the historical status, by which the targets managing task of satellite monitoring can efficiently accomplished, and which can be well applied in the field of city planning, management etc.

Key words: version management; ArcSDE; satellite monitoring; historical tracing